

Trabajo final de carrera: “Sistema de gestión de información de jugadores de baloncesto”

Memoria

Enrique Quiroga Rico

Proyecto para la gestión de información relativa a jugadores profesionales de baloncesto a nivel mundial, así como de sus estadísticas de juego y contratos.

Consultor: Àlex Caminals Sánchez de la Campa



AGRADECIMIENTOS

A mi esposa, por su infinita paciencia durante el tiempo que ha durado este proyecto.

RESUMEN

El presente trabajo final de carrera pretende plasmar los conocimientos relacionados con las bases de datos relacionales adquiridos a lo largo de los estudios de Ingeniería Técnica en Informática de Gestión.

Para ello se requiere el desarrollo completo de un proyecto, a partir de unos requerimientos proporcionados por el consultor (ejerciendo el rol de cliente). Éste comprende todas las fases habituales, desde el análisis de requerimientos, pasando por el diseño, la implementación y la validación.

Asimismo se ha documentado todo el trabajo realizado, con la finalidad de facilitar su comprensión por parte del consultor (cliente), así como de terceras partes que pudieran interactuar con el producto final, como por ejemplo desarrolladores de aplicaciones cliente.

El caso concreto implica la creación de una base de datos relacional, implementada sobre el sistema gestor Oracle 11g R2, que gestione la información relacionada con jugadores de baloncesto profesionales de todo el mundo.

El sistema debe incluir:

- Las tablas y demás elementos estructurales necesarios, como índices, disparadores, funciones, etc...para la correcta implementación del modelo de negocio planteado por los requerimientos del cliente.
- Procedimientos de alta, baja y modificación (ABM) para la manipulación de los datos relacionados con las tablas del modelo de negocio. Éstos se encargan de asegurar que los datos proporcionados al sistema sean adecuados, y de gestionar los posibles errores producidos durante su ejecución.
- Procedimientos de consulta que, a partir de unos parámetros de entrada, proporcionen una determinada información de salida, incluyendo datos calculados. Para ello se utilizan procedimientos almacenados. En algunas ocasiones se hace necesario el uso de funciones analíticas (caso de consultas del tipo “top N”).

- Tabla de auditoría para el registro de las llamadas a procedimientos almacenados, como los ABM y las consultas, incluyendo el nombre del procedimiento, la fecha y hora de la llamada, los parámetros de entrada y los de salida, incluyéndose en estos últimos los mensajes de gestión de excepciones.
- Módulo estadístico que, mediante el uso de tablas y disparadores adecuadamente diseñados, proporciona acceso inmediato, con el coste de una consulta simple, a indicadores relacionados con la población de datos incluida en el sistema.

ÍNDICE

ÍNDICE	4
ÍNDICE DE ILUSTRACIONES	5
ÍNDICE DE TABLAS	6
1. INTRODUCCIÓN.....	7
1.1. <i>Justificación.....</i>	7
1.2. <i>Objetivos.....</i>	7
1.3. <i>Enfoque y método seguido.....</i>	8
1.4. <i>Planificación del trabajo</i>	8
1.5. <i>Productos obtenidos.....</i>	14
1.6. <i>Contenido de la memoria</i>	15
2. ANÁLISIS DE REQUERIMIENTOS	16
2.1. <i>Requerimientos funcionales</i>	16
2.2. <i>Requerimientos no funcionales.....</i>	17
3. DISEÑO	19
3.1. <i>Diseño conceptual.....</i>	19
3.1.1. <i>Modelo de negocio.....</i>	19
3.1.2. <i>Módulo estadístico.....</i>	20
3.1.3. <i>Tabla de auditoría.....</i>	21
3.1.4. <i>Tabla de bajas lógicas</i>	21
3.1.5. <i>Decisiones de diseño</i>	22
3.1.6. <i>Validación</i>	24
3.2. <i>Diseño lógico.....</i>	25
3.2.1. <i>Transformación al modelo relacional</i>	26
3.2.2. <i>Restricciones de integridad</i>	27
3.2.3. <i>Decisiones de diseño</i>	32
3.2.4. <i>Validación</i>	33
3.3. <i>Diseño físico</i>	34
3.3.1. <i>Decisiones de diseño</i>	34
3.3.2. <i>Validación</i>	36
4. IMPLEMENTACIÓN.....	37
4.1. <i>Espacios de tabla.....</i>	37
4.2. <i>Usuarios.....</i>	37
4.3. <i>Tablas.....</i>	38
4.4. <i>Índices.....</i>	38
4.5. <i>Secuencias.....</i>	38
4.6. <i>Disparadores</i>	39
4.7. <i>Procedimientos ABM.....</i>	39

4.7.1.	IMPLEMENTACIÓN.....	40
4.7.2.	VALIDACIÓN.....	42
4.8.	<i>Procedimientos de relaciones especiales</i>	42
4.8.1.	IMPLEMENTACIÓN.....	42
4.8.2.	VALIDACIÓN.....	44
4.9.	<i>Disparadores de herencia disjunta</i>	44
4.9.1.	IMPLEMENTACIÓN.....	45
4.9.2.	VALIDACIÓN.....	45
4.10.	<i>Consultas</i>	45
4.10.1.	IMPLEMENTACIÓN.....	46
4.10.2.	VALIDACIÓN.....	49
4.11.	<i>Módulo estadístico</i>	49
4.11.1.	IMPLEMENTACIÓN.....	49
4.11.2.	VALIDACIÓN.....	54
5.	VALORACIÓN ECONÓMICA.....	54
5.1.	<i>ESTIMACIÓN INICIAL</i>	54
5.2.	<i>COSTE REAL</i>	55
6.	CONCLUSIONES	56
6.1.	<i>DIFICULTADES</i>	56
6.2.	<i>MEJORAS</i>	56
6.2.1.	CONTROL DE CONVOCATORIAS, ESTADÍSTICAS Y EQUIPOS ARBITRALES.....	56
6.2.2.	GRANULARIDAD DE LOS DISPARADORES ESTADÍSTICOS	57
6.2.3.	GESTIÓN DE BAJAS LÓGICAS.....	57
6.3.	<i>CONCLUSIONES</i>	57
7.	GLOSARIO	58
8.	BIBLIOGRAFIA	59
8.1.	<i>LIBROS</i>	59
8.2.	<i>WEBS MODELO DE NEGOCIO</i>	59
8.3.	<i>WEBS ÁMBITO TÉCNICO</i>	60
9.	ANEXOS	61

ÍNDICE DE ILUSTRACIONES

Ilustración 1:	temporización inicial de tareas según modelo de desarrollo.....	9
Ilustración 2:	temporización final de tareas según modelo de desarrollo	9
Ilustración 3:	diagrama de casos de uso	18
Ilustración 4:	diagrama Entidad-Relación.....	19

ÍNDICE DE TABLAS

Tabla 1: evolución de la planificación	14
Tabla 2: atributos de las entidades del modelo de negocio	20
Tabla 3: atributos de las entidades del módulo estadístico	21
Tabla 4: atributos de la tabla de auditoría	21
Tabla 5 : tabla de bajas lógicas	22
Tabla 6: validación del diseño conceptual	25
Tabla 7: transformación de entidades e interrelaciones al modelo relacional.....	26
Tabla 8: restricciones de integridad.....	32
Tabla 9: validación del diseño lógico	34
Tabla 10: índices creados	35
Tabla 11: validación diseño físico.....	36
Tabla 12: espacios de tabla.....	37
Tabla 13: índices creados	38
Tabla 14: secuencias.....	39
Tabla 15: procedimientos ABM.....	41
Tabla 16: procedimientos de relaciones críticas	43
Tabla 17: herencias.....	44
Tabla 18: disparadores control herencia disjunta	45
Tabla 19: procedimientos de consulta	48
Tabla 20: disparadores módulo estadístico	53
Tabla 21: detalle de los recursos humanos necesarios iniciales.....	54
Tabla 22: coste de los recursos humanos iniciales.....	55
Tabla 23: detalle de los recursos humanos necesarios reales	55
Tabla 24: coste de los recursos humanos reales.....	55

1. INTRODUCCIÓN

Mediante el presente apartado, se pretenden introducir los objetivos del proyecto, así como su desarrollo y el producto obtenido.

1.1. Justificación

El trabajo final de carrera, en adelante TFC, tiene como objetivo consolidar los conocimientos adquiridos por el alumno durante el ciclo formativo. Así, para un campo concreto, en este caso el de las bases de datos relacionales se propone un proyecto completo, similar al que se puede encontrar un profesional en el mundo empresarial. De esta manera se proporciona experiencia al alumno en cuanto a desarrollar un proyecto, y se consolidan y amplían sus conocimientos en relación a la temática escogida (a modo de especialización).

Hasta el momento, el alumno había obtenido conocimientos en las asignaturas relacionadas (BD y BDII) pero no había tenido oportunidad de realizar un proyecto completo, que englobara todas las fases del desarrollo (interpretación de requerimientos del cliente, diseño, implementación, validación y documentación).

1.2. Objetivos

Desde un punto de vista docente, el proyecto tiene como propósito consolidar y ampliar los conocimientos relacionados con bases de datos relacionales adquiridos por el diseñador durante la carrera. La metodología usada a tal fin ha consistido en el desarrollo completo de una base de datos, en adelante BD, que proporcionara respuesta a una necesidad concreta.

En relación con el producto, la finalidad del proyecto consistió en proporcionar un producto alineado con los requerimientos del cliente, así como de la documentación necesaria para su mantenimiento. Así mismo se aseguró que se cumplieran los criterios de calidad necesarios para el correcto diseño de una base de datos relacional:

- ✓ Normalización a la forma normal de Boyce-Codd, para evitar anomalías de diseño.
- ✓ Flexibilidad a cambios (escalabilidad de contenidos), con el fin de facilitar futuras ampliaciones del modelo.
- ✓ Independencia entre datos y procesos usuario, con el objetivo de simplificar la realización de cambios en el sistema.
- ✓ Integridad de los datos ante actualizaciones.

Dado que se desconoce el requisito de rendimiento de la BD, quedó fuera del objeto de este proyecto la optimización de la misma. Se estableció que el requerimiento consistiera en evitar ralentizaciones notables en su funcionamiento durante las consultas más complejas implementadas.

1.3. Enfoque y método seguido

El proyecto se ha desarrollado según el modelo de ciclo de vida en cascada para las etapas de análisis de requisitos y diseño. La decisión de usar dicha metodología se fundamentó en el hecho de que los requerimientos eran definitivos y no se producirían cambios importantes, y de que durante dichas etapas, disponer de una visión global del sistema facilitaría su comprensión.

Para las etapas de implementación y prueba se optó por el modelo iterativo e incremental basado en el ciclo de vida en cascada. Esta metodología permitió asegurar el correcto desempeño de las diferentes funcionalidades, entre las que habitualmente existía un grado de dependencia, a medida que se fueron implementando. De modo que aquellas que proveían de servicios a otras fueron validadas en primer lugar para evitar transferir errores a fases siguientes. Las validaciones se realizaron a través de juegos de pruebas que evaluaron la capacidad de cada funcionalidad para satisfacer los requisitos del cliente.

A lo largo del desarrollo del proyecto se efectuaron 3 entregas parciales conteniendo información actualizada sobre los avances realizados. La retroalimentación obtenida del cliente en cada una de ellas fue tomada en cuenta como nuevos requerimientos de cliente, lo cual provocó cambios en la planificación inicial del proyecto.

1.4. Planificación del trabajo

El proyecto se estructuró en etapas, determinadas por el modelo de desarrollo escogido. Su temporización coincidió, aproximadamente, con el del ciclo de vida en cascada, con la particularidad de que implementación y pruebas se planificaron conjuntamente.

Se adjuntan tablas mostrando la temporización, según el modelo de desarrollo, inicial y definitiva.

Nombre de tarea	Duración	Comienzo	Fin
Plan de trabajo	18 días	jue 28/02/13	dom 17/03/13
Análisis de requerimientos	8 días	lun 18/03/13	lun 25/03/13
Diseño	13 días	mar 26/03/13	dom 07/04/13
Implementación y pruebas	58 días	lun 08/04/13	mar 04/06/13
Entrega	7 días	mié 05/06/13	mar 11/06/13

Ilustración 1: temporización inicial de tareas según modelo de desarrollo

Nombre de tarea	Duración	Comienzo	Fin
Plan de trabajo	18 días	jue 28/02/13	dom 17/03/13
Análisis de requerimientos	8 días	lun 18/03/13	lun 25/03/13
Diseño	14 días	mar 25/03/13	dom 07/04/13
Implementación y pruebas	62 días	lun 08/04/13	mar 08/06/13
Entrega	4 días	mié 09/06/13	mar 12/06/13

Ilustración 2: temporización final de tareas según modelo de desarrollo

Las fases de implementación y pruebas se organizaron según las funcionalidades ofrecidas por el sistema, desde los componentes más generales hasta los más específicos.

La planificación inicial fue sufriendo modificaciones debido, principalmente, a mejoras en el diseño, detectadas durante las fases de implementación, o como aportación del cliente, y a dificultades a la implementación. Así, la planificación del proyecto fue revisada en 3 ocasiones para su ajuste a la realidad.

A continuación se comparan la planificación inicial y la final.

Planificación inicial					Planificación final				
Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
TFC-BBDD Relacionales	105 días	jue 28/02/13	mié 12/06/13		TFC-BBDD Relacionales	105 días	jue 28/02/13	mié 12/06/13	
Plan de trabajo	18 días	jue 28/02/13	dom 17/03/13		Plan de trabajo	18 días	jue 28/02/13	dom 17/03/13	
Análisis inicial de requerimientos	3 días	jue 28/02/13	sáb 02/03/13		Análisis inicial de requerimientos	3 días	jue 28/02/13	sáb 02/03/13	
Definición de objetivos	1 día	lun 04/03/13	lun 04/03/13	3	Definición de objetivos	1 día	lun 04/03/13	lun 04/03/13	3

Elección metodología de diseño	1 día	mar 05/03/13	mar 05/03/13	4	Elección metodología de diseño	1 día	mar 05/03/13	mar 05/03/13	4
Determinación de recursos	1 día	mié 06/03/13	mié 06/03/13	5	Determinación de recursos	1 día	mié 06/03/13	mié 06/03/13	5
Determinación de tareas	2 días	jue 07/03/13	vie 08/03/13	6	Determinación de tareas	2 días	jue 07/03/13	vie 08/03/13	6
Planificación de tareas	2 días	sáb 09/03/13	dom 10/03/13	7	Planificación de tareas	2 días	sáb 09/03/13	dom 10/03/13	7
Análisis de riesgos y plan de contingencias	1 día	lun 11/03/13	lun 11/03/13	8	Análisis de riesgos y plan de contingencias	1 día	lun 11/03/13	lun 11/03/13	8
Valoración coste proyecto	1 día	mar 12/03/13	mar 12/03/13	9	Valoración coste proyecto	1 día	mar 12/03/13	mar 12/03/13	9
Creación informe PAC1	4 días	mié 13/03/13	sáb 16/03/13	10	Creación informe PAC1	4 días	mié 13/03/13	sáb 16/03/13	10
Entrega informe PAC1 (plan de trabajo)	1 día	sáb 16/03/13	sáb 16/03/13	11	Entrega informe PAC1 (plan de trabajo)	1 día	sáb 16/03/13	sáb 16/03/13	11
Análisis de requerimientos	8 días	lun 18/03/13	lun 25/03/13	2	Análisis de requerimientos	8 días	lun 18/03/13	lun 25/03/13	2
Análisis de requerimientos funcionales	3 días	dom 17/03/13	mar 19/03/13	12	Análisis de requerimientos funcionales	3 días	dom 17/03/13	mar 19/03/13	12
Análisis de requerimientos no funcionales	3 días	mié 20/03/13	vie 22/03/13	14	Análisis de requerimientos no funcionales	3 días	mié 20/03/13	vie 22/03/13	14
Documentación fase de análisis	2 días	sáb 23/03/13	dom 24/03/13	15	Documentación fase de análisis	2 días	sáb 23/03/13	dom 24/03/13	15
Diseño	14 días	lun 25/03/13	dom 07/04/13	13	Diseño	14 días	lun 25/03/13	dom 07/04/13	13
Adquisición conocimientos Almacenes Datos	2 días	lun 25/03/13	mar 26/03/13	16	Adquisición conocimientos Almacenes Datos	2 días	lun 25/03/13	mar 26/03/13	16
Instalación, configuración y prueba herramienta CASE	1 día	mié 27/03/13	mié 27/03/13	18	Instalación, configuración y prueba herramienta CASE	1 día	mié 27/03/13	mié 27/03/13	18
Diseño conceptual	3 días	jue 28/03/13	sáb 30/03/13	19	Diseño conceptual	3 días	jue 28/03/13	sáb 30/03/13	19
Validación diseño conceptual	1 día	dom 31/03/13	dom 31/03/13	20	Validación diseño conceptual	1 día	dom 31/03/13	dom 31/03/13	20
Diseño lógico	3 días	lun 01/04/13	mié 03/04/13	21	Diseño lógico	3 días	lun 01/04/13	mié 03/04/13	21
Validación diseño lógico	1 día	mié 03/04/13	mié 03/04/13	22	Validación diseño lógico	1 día	mié 03/04/13	mié 03/04/13	22
Diseño físico	2 días	jue 04/04/13	vie 05/04/13	23	Diseño físico	2 días	jue 04/04/13	vie 05/04/13	23
Validación diseño físico	1 día	vie 05/04/13	vie 05/04/13	24	Validación diseño físico	1 día	vie 05/04/13	vie 05/04/13	24

Documentación Análisis y Diseño	2 días	sáb 06/04/13	dom 07/04/13	25	Documentación Análisis y Diseño	2 días	sáb 06/04/13	dom 07/04/13	25
Implementación y pruebas	58 días	lun 08/04/13	mar 04/06/13	17	Implementación y pruebas	62 días	lun 08/04/13	sáb 08/06/13	17
Instalación entorno SGBD	1 día	lun 08/04/13	lun 08/04/13	26	Instalación entorno SGBD	1 día	lun 08/04/13	lun 08/04/13	26
Script creación espacio de tablas	1 día	mar 09/04/13	mar 09/04/13	28	Script creación espacio de tablas	1 día	mar 09/04/13	mar 09/04/13	28
Implementación modelo de negocio	32 días	mié 10/04/13	sáb 11/05/13	29	Implementación modelo de negocio	32 días	mié 10/04/13	sáb 11/05/13	29
Implementación tablas modelo negocio	5 días	mié 10/04/13	dom 14/04/13	29	Implementación tablas modelo negocio	5 días	mié 10/04/13	dom 14/04/13	29
Script creación tablas modelo negocio	2 días	mié 10/04/13	jue 11/04/13	29	Script creación tablas modelo negocio	3 días	mié 10/04/13	vie 12/04/13	29
Pruebas tablas modelo negocio	2 días	vie 12/04/13	sáb 13/04/13	32	Pruebas tablas modelo negocio	2 días	sáb 13/04/13	dom 14/04/13	32
Introducción datos tablas modelo negocio	1 día	dom 14/04/13	dom 14/04/13	33	Script creación índices modelo negocio	1 día	lun 15/04/13	lun 15/04/13	31
Script creación índices modelo negocio	1 día	lun 15/04/13	lun 15/04/13	31	Implementación disparadores modelo negocio	3 días	mar 16/04/13	jue 18/04/13	31
Implementación disparadores modelo negocio	3 días	mar 16/04/13	jue 18/04/13	31	Script creación disparadores modelo negocio	2 días	mar 16/04/13	mié 17/04/13	34
Script creación disparadores modelo negocio	2 días	mar 16/04/13	mié 17/04/13	35	Pruebas disparadores modelo negocio	1 día	jue 18/04/13	jue 18/04/13	36
Pruebas disparadores modelo negocio	1 día	jue 18/04/13	jue 18/04/13	37	Documentación implementación modelo negocio	2 días	vie 19/04/13	sáb 20/04/13	37
Documentación implementación modelo negocio	2 días	vie 19/04/13	sáb 20/04/13	38	Entrega informe PAC2	1 día	sáb 20/04/13	sáb 20/04/13	38

Entrega informe PAC2	1 día	sáb 20/04/13	sáb 20/04/13	39	Implementación procedimientos modelo negocio	20 días	lun 22/04/13	sáb 11/05/13	35
Implementación procedimientos modelo negocio	20 días	lun 22/04/13	sáb 11/05/13	36	Adquisición conocimientos tratamiento excepciones	1 día	lun 22/04/13	lun 22/04/13	39
Adquisición conocimientos tratamiento excepciones	1 día	lun 22/04/13	lun 22/04/13	40	Script creación procedimientos ABM	6 días	mar 23/04/13	dom 28/04/13	41
Script creación procedimientos ABM	4 días	mar 23/04/13	vie 26/04/13	42	Pruebas procedimientos ABM	2 días	lun 29/04/13	mar 30/04/13	42
Pruebas procedimientos ABM	2 días	sáb 27/04/13	dom 28/04/13	43	Introducción datos en tablas modelo negocio	3 días	mié 01/05/13	vie 03/05/13	43
Script creación procedimientos estadísticas partidos	2 días	lun 29/04/13	mar 30/04/13	44	Script creación procedimiento estadísticas partidos	4 días	sáb 04/05/13	mar 07/05/13	44
Pruebas procedimientos estadísticas partidos	1 día	mié 01/05/13	mié 01/05/13	45	Pruebas procedimiento estadísticas partidos	1 día	mié 08/05/13	mié 08/05/13	45
Script creación procedimientos de consulta	4 días	jue 02/05/13	dom 05/05/13	46	Script creación procedimiento equipo arbitral	3 días	jue 09/05/13	sáb 11/05/13	46
Pruebas procedimientos estadísticas partidos	2 días	lun 06/05/13	mar 07/05/13	47	Pruebas procedimientos equipo arbitral	1 día	dom 12/05/13	dom 12/05/13	47
Documentación implementación procedimientos modelo negocio	4 días	mié 08/05/13	sáb 11/05/13	48	Mejora del sistema	3 días	lun 13/05/13	mié 15/05/13	48
Entrega informe PAC3	1 día	sáb 11/05/13	sáb 11/05/13	49	Creación disparadores control herencia	1 día	jue 16/05/13	jue 16/05/13	49
Implementación Módulo Estadístico	17 días	dom 12/05/13	mar 28/05/13	30	Pruebas disparadores control herencia	1 día	vie 17/05/13	vie 17/05/13	50
Script creación tablas módulo estadístico	3 días	dom 12/05/13	mar 14/05/13	49	Creación informe PAC3	2 días	sáb 18/05/13	dom 19/05/13	51

Pruebas tablas módulo estadístico	1 día	mié 15/05/13	mié 15/05/13	52	Entrega informe PAC3	1 día	dom 19/05/13	dom 19/05/13	52
Script creación disparadores módulo estadístico	2 días	jue 16/05/13	vie 17/05/13	53	Script creación procedimientos de consulta	9 días	lun 20/05/13	mar 28/05/13	53
Pruebas disparadores módulo estadístico	1 día	sáb 18/05/13	sáb 18/05/13	54	Pruebas procedimientos de consulta	2 días	mié 29/05/13	jue 30/05/13	54
Script creación procedimientos módulo estadístico	4 días	dom 19/05/13	mié 22/05/13	55	Documentación procedimientos de consulta	1 día	vie 31/05/13	vie 31/05/13	55
Pruebas procedimientos módulo estadístico	2 días	jue 23/05/13	vie 24/05/13	56	Implementación Módulo Estadístico	10 días	sáb 01/06/13	lun 10/06/13	30
Documentación implementación módulo estadístico	4 días	sáb 25/05/13	mar 28/05/13	57	Script creación disparadores módulo estadístico	6 días	sáb 01/06/13	jue 06/06/13	56
Implementación tabla auditoría para llamadas a procedimientos	7 días	mié 29/05/13	mar 04/06/13	51	Pruebas disparadores módulo estadístico	1 día	vie 07/06/13	vie 07/06/13	58
Script creación tabla auditoría	1 día	mié 29/05/13	mié 29/05/13	57	Documentación implementación módulo estadístico	1 día	sáb 08/06/13	sáb 08/06/13	59
Pruebas tabla auditoría	1 día	jue 30/05/13	jue 30/05/13	60	Entrega	4 días	dom 09/06/13	mié 12/06/13	60
Script creación procedimiento tabla auditoría	2 días	vie 31/05/13	sáb 01/06/13	61	Revisión memoria proyecto	2 días	dom 09/06/13	lun 10/06/13	60
Pruebas procedimientos tabla auditoría	1 día	dom 02/06/13	dom 02/06/13	62	Creación presentación proyecto	1 día	mar 11/06/13	mar 11/06/13	62
Documentación implementación tabla auditoría	2 días	lun 03/06/13	mar 04/06/13	63	Entrega final del proyecto	1 día	mié 12/06/13	mié 12/06/13	63
Entrega	7 días	mié 05/06/13	mar 11/06/13	59					
Revisión memoria proyecto	3 días	mié 05/06/13	vie 07/06/13	64					
Creación presentación proyecto	3 días	sáb 08/06/13	lun 10/06/13	66					

Entrega final del proyecto	1 día	mar 11/06/13	mar 11/06/13	67
----------------------------	-------	-----------------	-----------------	----

Tabla 1: evolución de la planificación

1.5. Productos obtenidos

Como resultado del desarrollo del proyecto se han obtenido una serie de productos que, en conjunción, consiguen que el sistema global cumpla los requerimientos del cliente.

Scripts para la creación y validación de todas las funcionalidades del sistema:

- Espacios de tabla. Se disponen diferentes espacios de tabla para los elementos del modelo de negocio (tablas, secuencias, disparadores...), para la totalidad de los índices del sistema y para las tablas auxiliares (log, bajas lógicas y módulo estadístico).
- Usuario, se crea el perfil del usuario administrador con los privilegios correspondientes para la implementación de todos los elementos del proyecto.
- Tablas del modelo de negocio, en las que se almacenarán las entidades básicas del sistema. Mediante el uso de restricciones de integridad se restringe el uso de valores no adecuados, así como el aseguramiento de las claves primarias y foráneas.
- Secuencias que proporcionan automáticamente diferentes valores a las claves primarias creadas.
- Disparadores de secuencia, se encargan de generar un nuevo valor de secuencia, como clave primaria, cada vez que se introduce un nuevo elemento.
- Disparadores de aseguramiento de herencia disjunta. Se encargan de asegurar que los elementos definidos dentro de una relación de herencia disjunta cumplan con tales requisitos.
- Procedimientos de alta, baja y modificación (ABM). Se encargan de asegurar que los datos proporcionados al sistema sean adecuados, y de gestionar los posibles errores producidos durante su ejecución.
- Tablas del módulo estadístico. Contienen la información relacionada con los indicadores correspondientes, debidamente tratada y actualizada.
- Disparadores del módulo estadístico. Transforman, ante la presencia de un evento, la información contenida en las tablas del modelo de negocio y actualizan el contenido de las tablas contenedoras de indicadores estadísticos.
- Tabla auxiliar de auditoría, contiene el registro de las llamadas a procedimientos almacenados.
- Tabla auxiliar de bajas lógicas. Guarda información relativa a los elementos en baja lógica.

- Población de datos iniciales, para todas las tablas del modelo de negocio.
- Juegos de pruebas para todos los elementos definidos anteriormente (tablas, disparadores, secuencias, procedimientos almacenados, módulo estadístico).

Documentación asociada:

- Resultado de los juegos de pruebas, como evidencia del correcto funcionamiento del sistema.
- Diagramas de planificación del proyecto, en los que se puede apreciar la evolución del proyecto y los cambios sufridos por éste.
- Instrucciones de instalación de la base de datos, incluyendo los juegos de pruebas.
- Memoria en la que se describe la ejecución del proyecto durante la totalidad de su desarrollo, incluyendo la justificación de las decisiones de diseño, coste del proyecto y conclusiones.
- Presentación que sintetiza el desarrollo del proyecto.

1.6. Contenido de la memoria

En los siguientes puntos de la presente memoria se describirán los detalles de la ejecución del proyecto, de acuerdo al modelo de desarrollo escogido, así como otras informaciones relacionadas con el proyecto.

Principalmente se describirán las siguientes etapas:

- Análisis de requerimientos, tanto funcionales como no funcionales, a partir de los requerimientos iniciales del cliente.
- Diseño, conceptual, físico y lógico del sistema, así como la argumentación de las decisiones de diseño tomadas y su validación contra los requerimientos definidos en la etapa anterior.
- Implementación de la base de datos y validación mediante juegos de pruebas.
- Valoración económica del proyecto, comparando previsión inicial y coste real del mismo.
- Conclusiones del proyecto, incluyendo dificultades halladas y posibles mejoras.
- Glosario
- Bibliografía
- Anexos

2. ANÁLISIS DE REQUERIMIENTOS

Durante la fase de análisis de requerimientos se trató de interpretar las necesidades del cliente a partir de la información proporcionada, para traducirlas a especificaciones técnicas. Para aquellos requisitos no especificados se optó por consultar al cliente o aplicar el sentido común.

Cabe destacar que, como resultado de esta actividad se obtuvieron dos tipos diferentes de requerimientos. Los funcionales, que describen los servicios que el sistema deberá proporcionar a sus usuarios, y los no funcionales, que recogen todas aquellas limitaciones derivadas del entorno, metodología y tecnología presentes en el proyecto.

2.1. Requerimientos funcionales

Funciones del sistema:

- Almacenar toda la información necesaria para la gestión de jugadores, así como de sus contratos y estadísticas.
- Contemplar la carrera deportiva completa de cada jugador.
- Incluir los datos relativos a cada jugador.
 - Guardar, como mínimo, el estado médico actual.
- Incluir los datos relativos a los equipos.
- Guardar algunos datos relacionados con los contratos de los jugadores con sus equipos.
 - Registrar los datos relativos a transacciones económicas, como mínimo, en euros.
 - Establecer que la duración de los contratos sea de años completos.
 - Establecer que los años contractuales comiencen el 1 de julio y finalicen el 30 de junio.
- Almacenar las estadísticas por partido de cada jugador.
 - Calcular automáticamente una valoración de cada jugador a partir de otras estadísticas de cada partido.
 - Calcular una valoración ponderada de cada jugador en función de la competición a la que pertenezca el partido, y a partir del dato de valoración del jugador.

2.2. Requerimientos no funcionales

El proyecto debía:

- Desarrollarse según el modelo relacional de bases de datos.
 - Normalizar la base de datos a la forma normal de Boyce-Codd
 - Asegurar su flexibilidad ante cambios (escalabilidad de contenidos).
 - Garantizar la independencia entre datos y procesos usuario
 - Guardar la integridad de los datos ante actualizaciones.
- No contemplar el desarrollo de una interface gráfica.
- Desarrollar la base de datos como una aplicación cliente/servidor.
- Asegurar un rendimiento del sistema tal que asegure la ausencia de ralentizaciones notables, que pudieran resultar molestas al usuario medio, en aquellas consultas de mayor complejidad.
- Garantizar la compatibilidad con la familia de sistemas operativos Windows de 32bits. Aun no existiendo ninguna especificación en este sentido, se decide su uso por tratarse del más extendido en el mercado.
- Reflejar el modelo conceptual del diseño mediante un diagrama E/R o UML.
- Listar todas las restricciones de integridad halladas.
- Crear los scripts necesarios para la implementación de los elementos que formen parte del sistema.
- Asegurar que los procedimientos almacenados:
 - Devuelven un parámetro de tipo cadena de texto que informará del éxito de su ejecución.
 - Tratan las excepciones.
- Implementar una tabla de log en la que se registren todas las llamadas a procedimientos. Se almacenará el nombre del procedimiento, los parámetros de entrada y los de salida.
- Documentar correctamente los procedimientos almacenados para facilitar su entendimiento por parte de los programadores de la capa de presentación.
 - Describir el funcionamiento de cada procedimiento a alto nivel.
 - Definir el dominio de los parámetros de entrada.
 - Especificar el dominio de los parámetros de salida, incluyendo los códigos de error y su interpretación.
- Insertar comentarios en el código de los procedimientos que describan su funcionamiento interno para facilitar su mantenimiento.
- Inicializar la base de datos con un conjunto de datos suficiente, de manera que todas las relaciones dispongan de registros que simulen fielmente la realidad representada por el modelo.

- Proveer de un juego de pruebas que garantice el correcto funcionamiento del sistema, incluyendo el control de errores y excepciones, para todos los requerimientos funcionales establecidos.
- Utilizar el sistema gestor de bases de datos Oracle.
- Proporcionar los privilegios adecuados a cada perfil de usuario, de acuerdo a lo requerido por las especificaciones. Así la Asociación Mundial de Jugadores dispondrá de privilegios para la ejecución de procedimientos almacenados (ABM, módulo estadístico o consultas); mientras que equipos y federaciones dispondrán de permiso únicamente para la consulta de información a través de los procedimientos habilitados a tal fin (consultas y módulo estadístico). El administrador de la base de datos dispondrá de todos los privilegios.

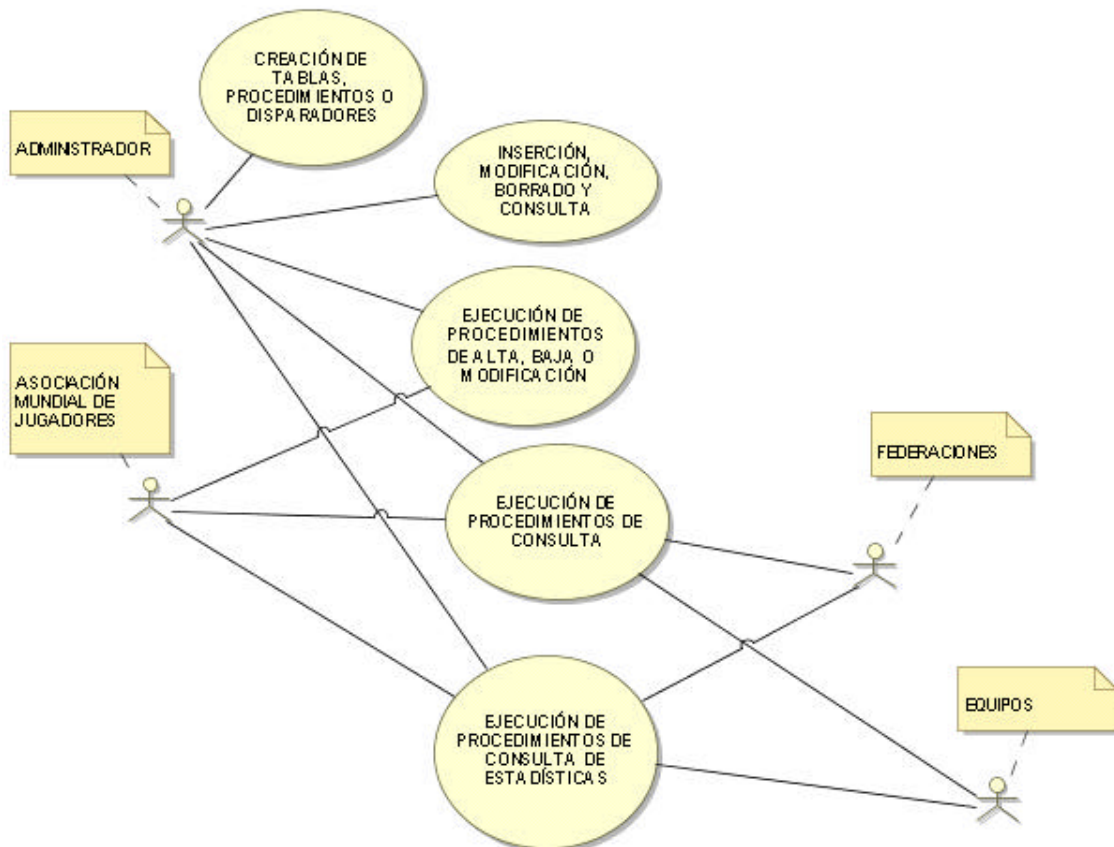


Ilustración 3: diagrama de casos de uso

- Proporcionar una política de copias de seguridad tal que asegure la recuperabilidad de los datos en función de la frecuencia con la que el sistema sufre modificaciones.

3. DISEÑO

El diseño de la base de datos se llevó a cabo en tres etapas diferenciadas (diseño conceptual, lógico y físico). De cada una de ellas se obtuvo un resultado que constituyó el punto de partida de la siguiente.

3.1. Diseño conceptual

Durante la fase de diseño conceptual se obtuvo una estructura de la información representativa de la solución a implementar según los requerimientos funcionales proporcionados por el cliente. Esta fase no estuvo condicionada por la tecnología a utilizar, así el resultado fue común independientemente del modelo de base de datos a implementar.

3.1.1. Modelo de negocio

Mediante el diagrama Entidad-Relación (en adelante diagrama E/R) se muestran las entidades del modelo de negocio, así como sus interrelaciones y la conectividad de éstas, definidas como solución a los requerimientos del cliente.

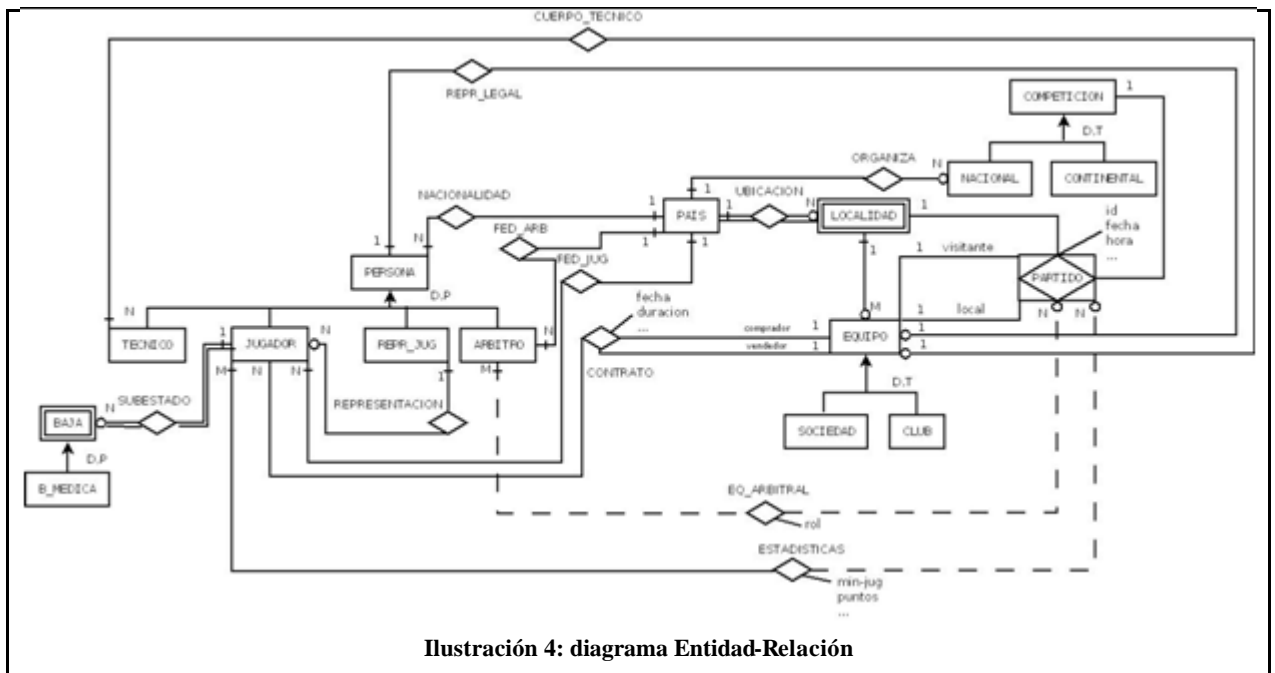


Ilustración 4: diagrama Entidad-Relación

A partir del diagrama E/R se definieron los atributos, incluyendo la clave primaria (subrayada), de cada una de las entidades. Los atributos derivados de las interrelaciones entre entidades se definieron durante la etapa de diseño lógico.

Entidad	Atributos	Observaciones
PERSONA	<u>id-persona</u> , dni, nombre-apellidos, fecha-nacimiento	
JUGADOR	<u>id-persona</u> , genero, modalidad, num-federado, altura, peso, posición, url-web, url-video, actividad	Entidad subclase de Persona.
ARBITRO	<u>id-persona</u>	Entidad subclase de Persona.
REPR_JUG	<u>id-persona</u> , teléfono, email	Entidad subclase de Persona.
TECNICO	<u>id-persona</u> , cargo	Entidad subclase de Persona.
BAJA	<u>expediente</u> , inicio, final-estimado, final-real	
B_MEDICA	<u>expediente</u> , diagnostico	Entidad subclase de Baja.
PAIS	<u>nombre</u> , continente	
LOCALIDAD	<u>id-localidad</u> , municipio, provincia	Entidad débil de País.
EQUIPO	<u>id-equipo</u> , nombre, direccion, telefono, url-web, estatus	
SOCIEDAD	<u>id-equipo</u> , tipo-sociedad	Entidad subclase de Equipo.
CLUB	<u>id-equipo</u> , num-socios	Entidad subclase de Equipo
COMPETICION	<u>id-comp</u> , nombre, año-ini, duracion, genero, modalidad, categoria, ponderacion,	
NACIONAL	<u>id-comp</u>	Entidad subclase de Competicion.
CONTINENTAL	<u>id-comp</u> , continente	Entidad subclase de Competicion.

Tabla 2: atributos de las entidades del modelo de negocio

3.1.2. Módulo estadístico

A partir de los requerimientos de cliente referentes al módulo estadístico se definieron los atributos para cada una de sus relaciones. El proceso consistió en determinar los atributos, para cada una de las consultas que forman el módulo estadístico, que satisficieran completamente las especificaciones dadas por el cliente.

En la tabla adjunta se muestran los atributos de cada entidad, las claves principales se muestran subrayadas. En la columna “Origen” se detallan el atributo y la entidad, entre paréntesis, del que procede cada uno de ellos. En el caso de atributos calculados se indica dicha condición.

Entidad	Atributos	Origen
DW_ACTIVOS	<u>genero</u> <u>modalidad</u> num_jug_activos	genero (JUGADOR) modalidad (JUGADOR) atributo calculado

DW_ANOTADOR	<u>competicion</u> , año_comp <u>id_jug</u> nombre_jug puntos	nombre (COMPETICION) año_ini (COMPETICION) id_persona(JUGADOR) nombre_y_apellidos (PERSONA) atributo calculado
DW_SALARIO_MAX	genero modalidad <u>id_jug</u> nombre_jug puntos	genero (JUGADOR) modalidad (JUGADOR) id_persona(JUGADOR) nombre-apellidos (JUGADOR) atributo calculado
DW_FORTUNA	<u>id-jug</u> nombre_jug ganancias	id_persona(JUGADOR) nombre-apellidos (JUGADOR) atributo calculado
DW_SUELDO_MEDIO	<u>pais</u> <u>temporada</u> <u>genero</u> <u>modalidad</u> sueldo_medio	nombre (PAIS) año-ini (COMPETICION) genero (COMPETICION) modalidad (COMPETICION) atributo calculado
DW_MEJORES_EQUIPOS	<u>id-comp</u> competicion temporada <u>id-eq</u> nombre_eq <u>puntos</u>	id_comp (COMPETICION) nombre(COMPETICION) año-ini (COMPETICION) id_equipo (EQUIPO) nombre (EQUIPO) atributo calculado
DW_MEJOR_JUGADOR	<u>genero</u> <u>modalidad</u> año <u>id-jug</u> nombre-jug media_val_pond	genero (JUGADOR) modalidad (JUGADOR) año_ini(COMPETICION) id_persona(JUGADOR) nombre_apellidos (JUGADOR) atributo calculado

Tabla 3: atributos de las entidades del módulo estadístico

3.1.3. Tabla de auditoría

A fin de controlar todas las llamadas a procedimientos, y sus resultados, se crea una tabla de auditoría. Al no depender directamente de ninguna entidad, ni interrelación, únicamente se detallan los atributos que la componen.

Entidad	Atributos
LOG_PROCS	<u>id-log</u> , fecha- hora, procedimiento, par-ent, par-sal

Tabla 4: atributos de la tabla de auditoría

3.1.4. Tabla de bajas lógicas

Las bajas de objetos tienen como finalidad, en el modelo creado, anular su intervención en cálculos o consultas, manteniendo su existencia dentro de la base de datos. De esta manera se

asegura la reversibilidad de la baja. Se ha de valorar el hecho de que el modelo presenta pocas necesidades para la anulación de objetos, debido a su carácter de registro histórico. La base de datos registra hechos ocurridos y no será habitual dar de baja registros.

Las bajas lógicas se realizarán mediante el registro del tipo de entidad, el identificador de la instancia y la fecha en la que se realiza el registro en una tabla especialmente habilitada a tal efecto. La tabla se ubica en el espacio de tabla “DW_LOG”, para evitar manipulaciones. Mediante este método se consigue un coste inferior al de un atributo de estado en cada relación, ya que para comprobar si un objeto se halla deshabilitado únicamente habrá que buscar en una tabla de pequeñas dimensiones y poco poblada. Además se ahorra espacio al no tener que disponer de un atributo dedicado para cada entidad, esté o no en baja lógica.

Entidad	Atributos
BAJA_LOGICA	<u>tabla</u> , <u>elemento</u> , fecha- hora

Tabla 5 : tabla de bajas lógicas

El atributo “elemento” se definió como VARCHAR a fin de evitar incompatibilidades entre los diferentes formatos de clave primaria existentes en el sistema, “NUMBER” y “VARCHAR2”. En el caso de bajas realizadas manualmente, mediante la sentencia “INSERT”, el usuario ya se encargará de realizar la conversión de tipos, en el caso de claves primarias de tipo “NUMBER”. Para los casos en los que exista procedimiento de baja, el gestor de base de datos se encarga de realizar una conversión implícita. De esta manera, todas las relaciones del sistema pueden registrarse en la tabla de bajas lógicas.

3.1.5. Decisiones de diseño

A continuación, y para un mejor entendimiento del modelo definido durante la etapa de diseño conceptual, se justifican las decisiones de desarrollo que pueden resultar menos evidentes:

- La entidad “Persona” permite disponer de un conjunto de atributos mínimos para todos los recursos humanos involucrados en el modelo. De esta manera se facilita la especialización, requerida por las singularidades propias de cada subclase, y la posibilidad de ampliar el sistema con nuevas entidades (roles) o atributos (funcionalidades). Los roles sin requerimientos específicos, como los representantes legales de los equipos, quedan englobados en la superclase Persona.
- Se establece el atributo secuencial “id-persona” de “Persona” a fin de establecer una clave primaria de un solo atributo, en lugar de “dni” + “nacionalidad”, ya que la adopción del DNI como clave primaria única podría provocar algún problema de repetición de valores entre diferentes países.
- El atributo “altura” de la entidad “Jugador” hace referencia, independientemente de la modalidad a la que esté adscrito, a la altura total de la persona. En el mismo sentido, el

atributo “peso” no contempla el peso de la silla de ruedas en el caso de la modalidad para jugadores con discapacidad física.

- Se incorporan los atributos “genero” y “modalidad”, ya existentes en “Competición”, a la entidad “Jugador” para conseguir que las consultas que los incorporen resulten más directas. También posibilitan, de otra manera sería imposible, el control en la asignación de jugadores a las competiciones adecuadas.
- La entidad “Localidad” requiere de “País” para quedar completamente definida. Por tanto se ha definido como entidad débil. El atributo “Provincia” acaba de definir el municipio dentro de un país, ya que puede darse el caso de hallar municipios con el mismo nombre dentro del mismo país. Mediante el atributo “id_localidad”, código autonómico, se identifica claramente al municipio, sin necesidad de recurrir a la provincia y al país. Se descarta el uso del código postal debido a la existencia de localidades que comparten el mismo código, y a la ausencia de codificación postal en algunos países (como Irlanda).
- La entidad “Baja” se ha definido como débil, ya que necesita de “Jugador” para quedar completamente definida.
- A fin de estandarizar las diferentes duraciones de los distintos campeonatos, se han habilitado los atributos “año-ini”, que indica el año de inicio, y “duración”, que contempla el número de años por los que se extiende el campeonato. Esta estructura proporciona mayor flexibilidad a la hora de acoger competiciones con diferentes formatos, pudiendo contemplar eventos de duración superior a la temporada (por ejemplo torneos clasificatorios para el campeonato mundial). Se controlará que el año de finalización no sea anterior al de inicio.
- Las competiciones mundiales, como los Juegos Olímpicos y el Campeonato Mundial de Selecciones, quedan incluidas en las continentales, al no existir diferencias entre ambas, ya que comparten número de árbitros. Se registrarán con el valor “mundial”.
- En aquellas entidades en las que la clave primaria está compuesta por varios atributos (“Competicion”, “Persona”, “Localidad”, “Equipo” y “Log_procs”) se ha habilitado una clave candidata que actúa como clave primaria. De esta manera se aligeran las claves foráneas y consultas que se relacionen con las entidades afectadas. Para asegurar la ausencia de repetición en el resto de atributos, se controlará la unicidad en los atributos que conforman la clave primaria natural.
- Se desestima la creación de una entidad “Continente”, debido a la ausencia de información de interés relacionada con la misma. Así, el nombre del continente organizador quedará recogido en el atributo “continente” de la entidad “Continental” (subclase de “Competición”).

- El atributo “cargo” de la entidad “Tecnico” especifica el cargo del técnico dentro del equipo. Algunos valores válidos podrían ser: entrenador, segundo entrenador, auxiliar, utillero, masajista, médico, psicólogo deportivo... Aunque que cada club puede disponer de un cuerpo técnico con una estructura propia. Por tanto se decide no restringir sus valores válidos y ofrecer flexibilidad ante distintos cargos.
- Se ha decidido que la entidad “Partido” se implemente por medio de una entidad asociativa. De este modo se permite su conexión con las interrelaciones “Eq_Arbital” y “Estadísticas”.
- Los atributos “est_convocatoria “ y “est_eq_arb” de la entidad “Partido” se encargan de mostrar el nivel de información registrada acerca de las interrelaciones “Eq_arbital” y “Estadísticas” en relación a cada instancia. De esta manera se podrá diferenciar entre partidos no disputados aún, y por tanto sin equipo arbitral y sin estadísticas, y los históricos. Sobre estos últimos se mostrará la información pendiente de introducir.
- Para asegurar que las relaciones de generalización/especialización definidas cumplan su condición de disjuntas, tal y como se ha especificado en su diseño, se deberá implementar un sistema de control mediante disparadores y procedimientos almacenados que comprueben que la instancia a crear no exista bajo otro rol hermano.
- Los siguientes atributos, debido a que disponen de un listado cerrado de valores válidos, deberán ser controlados para evitar errores:
 - “modalidad”, “genero”, “actividad” y “estado”, de la entidad “Jugador”.
 - “continente”, de la entidad “Continental”.
 - “categoría”, “modalidad” y “genero”, de la entidad “Competicion”.

3.1.6. Validación

Durante la validación se comprobó que el resultado de la fase, en este caso la del diseño conceptual, cumpliera con los requerimientos aplicables recogidos durante la etapa de definición de requerimientos funcionales.

A continuación se muestra todos los conceptos verificados:

Requerimiento	Resultado
Conceptos incluidos en los requisitos funcionales.	El diseño conceptual ha integrado todos los conceptos incluidos en los requisitos funcionales, entre entidades e interrelaciones.
Información mínima requerida.	Se han incluido todos los atributos directamente relacionados con entidades. Aquellos relacionados con interrelaciones se han anticipados para evitar errores, aunque serán incluidos durante la fase del diseño lógico.

Integridad de unicidad de la clave primaria.	Todas las entidades disponen de una clave primaria que identifica inequívocamente a cada instancia.
Integridad del dominio.	Se ha observado la necesidad de controlar aquellos atributos que disponen de un conjunto limitado de valores válidos, y que no pueden ser verificados únicamente mediante la pertenencia al dominio del atributo.
Normalización a FNBC.	Ninguna entidad muestra la existencia de varias claves candidatas compuestas que compartan algún elemento.
Flexibilidad.	Las entidades se han desarrollado contemplando posibles futuras ampliaciones en el sistema; siempre y cuando este hecho no haya ido en detrimento de la claridad y calidad del diseño. Se ha asegurado que aquellos conceptos que pudieran presentar falta de uniformidad entre instancias presenten flexibilidad suficiente para adaptarse a cualquier caso. Un ejemplo lo constituyen los atributos “año-ini” y “duración” de la entidad “Competición”.
Eficiencia	Se han simplificado, en lo posible, las claves primarias mediante atributos identificadores, a fin de mejorar el rendimiento de la base de datos. Ya que de este modo se consiguen búsquedas más rápidas.

Tabla 6: validación del diseño conceptual

3.2. Diseño lógico

El cometido de esta fase fue adaptar el resultado de la fase anterior a la tecnología concreta a utilizar, en este caso al modelo relacional de bases de datos.

La transformación consistió en:

- Convertir las entidades definidas durante la fase anterior en relaciones.
- Cambiar las interrelaciones identificadas por claves foráneas en relaciones ya conocidas, o crear nuevas relaciones.

Cabe destacar que la transformación de interrelaciones únicamente afectó a las entidades e interrelaciones del modelo de negocio, ya que, tanto el módulo estadístico como la tabla de auditoría y la de bajas lógicas carecen de éstas. Por tanto, sólo se contempló la transformación de entidades e interrelaciones del modelo de negocio.

3.2.1. Transformación al modelo relacional

En las tablas siguientes se muestran las relaciones resultantes de la adaptación al modelo relacional, así como indicación de las claves foráneas y su relación de origen. En rojo se muestran las entidades que, por su relativa importancia o baja implicación en el sistema, se consideró que pudieran presentar valor “*null*”.

Entidad	Atributos	Claves foráneas
PERSONA	<u>id-persona</u> , dni, nombre-apellidos, fecha-nacimiento, nacionalidad	nacionalidad (PAIS)
JUGADOR	<u>id-persona</u> , genero, modalidad, num-federado, altura, peso, posición, url-web , url-video , actividad, representante, fed-jug	representante (REPR_JUG) fed-jug (PAIS)
ARBITRO	<u>id-persona</u> , fed-arb	fed-arb (PAIS)
REPR_JUG	<u>id-persona</u> , teléfono, email	
TECNICO	<u>id-persona</u> , cargo, equipo	equipo (EQUIPO)
BAJA	<u>expediente</u> , jugador, inicio, final-estimado, final-real	jugador (JUGADOR)
B_MEDICA	<u>expediente</u> , diagnostico	
PAIS	<u>nombre</u> , continente	
LOCALIDAD	<u>id-localidad</u> , municipio, provincia, país	pais (PAIS)
EQUIPO	<u>id-equipo</u> , nombre, direccion, telefono, url-web , repr-legal, localidad	repr-legal (PERSONA) localidad (LOCALIDAD)
SOCIEDAD	<u>id-equipo</u> , tipo-sociedad	
CLUB	<u>id-equipo</u> , num-socios	
COMPETICION	<u>id-comp</u> , nombre, año-ini, duracion, genero, modalidad, categoria, ponderacion,	
NACIONAL	<u>id-comp</u> , país	pais (PAIS)
CONTINENTAL	<u>id-comp</u> , continente	
PARTIDO	<u>id-partido</u> , fecha-hora, pabellon, municipio, eq-local, eq-visitante, competicion, est-convocatoria , est-eq-arb	municipio (LOCALIDAD) competicion (COMPETICION) eq-local (EQUIPO) eq-visitante (EQUIPO)
CONTRATO	<u>id-contrato</u> , jugador, eq-comprador, eq-vendedor, fecha-firma, fecha-ini, fecha-fin, salario, compensación, valor-operacion	jugador (JUGADOR) eq-comprador (EQUIPO) eq-vendedor (EQUIPO)
EQ_ARBITRAL	<u>id-arbitraje</u> , partido, arbitro, rol	partido (PARTIDO) arbitro (ARBITRO)
ESTADISTICAS	<u>id-estadistica</u> , jugador, partido, min-jug, puntos, t1i, t1e, t2i, t2e, t3i, t3e, rd, ro, ast, tf, tc, pr, pp, fc, fr, valoracion , valor-ponderada	jugador (JUGADOR) partido (PARTIDO)

Tabla 7: transformación de entidades e interrelaciones al modelo relacional

Los atributos “valoración” y “val-ponderada” de la relación “Estadísticas” obtienen sus valores a partir de cálculos realizados sobre otros atributos de la misma relación.

3.2.2. Restricciones de integridad

Mediante las restricciones de integridad se garantiza la consistencia y la corrección de los datos almacenados en la base de datos.

A continuación se detallan las restricciones de chequeo de validez, integridad de entidad, integridad referencial y unicidad, aplicadas a los atributos de las relaciones definidas durante la transformación de las entidades e interrelaciones al modelo relacional. La restricción de obligatoriedad (atributos que no pueden permanecer vacíos o “null”) ya se mostró en la tabla 5 del apartado anterior.

Relación	Atributos	Restricción	Motivo
Pais	nombre	clave primaria	Definir completamente a la entidad y evita repetición de instancias.
	continente	Validación del contenido	Restringir los valores válidos al modelo de 5 continentes (“Africa”, “America”, “Asia”, “Europa” y “Oceania”).
Localidad	id_localidad	Clave primaria	Definir completamente a la entidad, evitar repetición de instancias y simplifica la clave primaria a un único atributo.
	pais	Clave foránea	Relacionar a “Localidad” con “País”, a través del atributo (clave primaria para “Pais”).
	municipio, provincia, pais	Unicidad	Evitar la repetición de instancias con diferentes claves primarias, es necesario añadir esta restricción para garantizar la ausencia de repeticiones de instancias.
Competicion	id_comp	Clave primaria	Definir a la entidad y simplificar la clave primaria a un único atributo.
	nombre, año_ini	Unicidad	Evitar la repetición de instancias con diferentes claves primarias, es necesario añadir esta restricción para garantizar la ausencia de repeticiones de instancias.
	año_ini	Validación del contenido	Asegurar que el año de inicio sea un valor coherente, entre 1900 y 2100.
	duracion	Validación del contenido	Comprobar que la duración del campeonato sea coherente con la realidad. Se establece que la duración máxima de una competición sea de 4 años, que corresponde con la fase clasificatoria de un mundial o un continental.
	genero	Validación del contenido	Verificar que el género tenga un valor coherente (“masculino” o “femenino”).

	modalidad	Validación del contenido	Asegurar que la modalidad escogida esté contemplada por el modelo diseñado (“normal” o “silla de ruedas”).
	categoria	Validación del contenido	Garantizar que las categorías recogidas sean “1ª” o “2ª”.
	ponderacion	Validación del contenido	Comprobar que la ponderación adquiera los valores “8” o “10”.
Nacional	id_comp	Clave primaria	Definir a la entidad y simplificar la clave primaria a un único atributo.
	Id_comp	Clave foránea	Desplegar la relación de generalización / especialización con su superclase “Competicion”.
	pais	Clave foránea	Relacionar a “Nacional” con “País”, a través del atributo (clave primaria para “País”).
Continental	id_comp	Clave primaria	Definir a la entidad y simplificar la clave primaria a un único atributo.
	id_comp	Clave foránea	Desplegar la relación de generalización / especialización con su superclase “Competicion”.
	continente	Validación del contenido	Restringir los valores válidos al modelo de 5 continentes (“Africa”, “America”, “Asia”, “Europa” y “Oceania”).
Persona	id_persona	Clave primaria	Definir completamente a la entidad y evitar repetición de instancias. Corresponde al documento nacional de identidad.
	nacionalidad	Clave foránea	Relacionar a “Persona” con “País”, a través del atributo (clave primaria para “País”).
Arbitro	id_persona	Clave primaria	Definir completamente a la entidad y evitar repetición de instancias. Corresponde al documento nacional de identidad
	id_persona	Clave foránea	Desplegar la relación de generalización / especialización con su superclase “Persona”.
	fed_arb	Clave foránea	Relacionar a “Arbitro” con “País”, a través del atributo (clave primaria para “País”).
Repr_jug	id_persona	Clave primaria	Definir completamente a la entidad y evitar repetición de instancias. Corresponde al documento nacional de identidad.
	id_persona	Clave foránea	Desplegar la relación de generalización / especialización con su superclase “Persona”.
	email	Validación del contenido	Asegurar que el atributo contiene una dirección de email.
	telefono	Unicidad	Evitar la repetición del atributo en diversas instancias. Se determina que cada representante debe disponer de un nº de teléfono propio.
	email	Unicidad	Evitar la repetición del atributo en diversas instancias. Se determina que cada representante debe disponer de una dirección de email propia.

Jugador	id_persona	Clave primaria	Definir completamente a la entidad y evita repetición de instancias. Corresponde al documento nacional de identidad.
	id_persona	Clave foránea	Desplegar la relación de generalización / especialización con su superclase "Persona".
	representante	Clave foránea	Relacionar a "Jugador" con "Representante", a través del atributo (clave primaria para "Representante").
	fed_jug	Clave foránea	Relacionar a "Jugador" con "Pais", a través del atributo (clave primaria para "Pais").
	num_federado, fed_jug	Unicidad	Asegurar que el número de federado de cada jugador sea único para cada país.
	genero	Validación del contenido	Verificar que el género tenga un valor coherente ("masculino" o "femenino").
	modalidad	Validación del contenido	Asegurar que la modalidad escogida esté contemplada por el modelo diseñado ("normal" o "silla de ruedas").
	altura	Validación del contenido	Restringir el rango de alturas válidas a valores reales (entre 1.60 y 2.50m).
	peso	Validación del contenido	Restringir el rango de pesos válidos a valores reales (entre 50 y 150Kg).
	posicion	Validación del contenido	Comprobar que el valor se halle entre las posiciones reales ("base", "escolta", "alero", "ala-pivot" y "pivot").
	url_web	Unicidad	Evitar la repetición del atributo en diversas instancias. Se determina que cada jugador debe disponer de una url_web propia.
	url_video	Unicidad	Evitar la repetición del atributo en diversas instancias. Se determina que cada jugador debe disponer de una url de video promocional propia.
actividad	Validación del contenido	Asegurar que el valor de actividad se halle entre los dos valores posibles ("activo" o "retirado").	
Baja	expediente	Clave primaria	Definir completamente a la entidad y evitar repetición de instancias. Se trata de un atributo real, generado por los servicios médicos, por lo que aunque pudieran repetirse los valores del resto de atributos, se trataría de dos instancias diferentes. No es un atributo identificador creado únicamente para simplificar la clave primaria.
	jugador	Clave foránea	Relacionar a "Baja" con "Jugador", a través del atributo (clave primaria para "Jugador").
	jugador, inicio	Unicidad	Evitar la repetición de los atributo en diversas instancias. Se determina que cada jugador puede tener una única baja abierta en un día.
	final_estimado, inicio	Validación del contenido	Garantizar que la fecha final estimada sea posterior a la fecha de inicio.
	final_real, inicio	Validación del contenido	Garantizar que la fecha final real sea posterior a la fecha de inicio.

B_Medica	expediente	Clave primaria	Definir completamente a la entidad y evitar repetición de instancias. Se trata de un atributo real, por lo que aunque pudieran repetirse los valores del resto de atributos, se trataría de dos instancias diferentes. No es un atributo identificador creado para simplificar la clave primaria.
	expediente	Clave foránea	Desplegar la relación de generalización / especialización con su superclase “Baja”.
Equipo	id_equipo	Clave primaria	Definir a la entidad y evitar repetición de instancias. Atributo real que se corresponde con el nº de identificación fiscal o similar.
	repr_legal	Clave foránea	Relacionar a “Equipo” con “Persona”, a través del atributo (clave primaria para “Persona”).
	localidad	Clave foránea	Relacionar a “Equipo” con “Localidad”, a través del atributo (clave primaria para “Localidad”).
	nombre, direccion	Unicidad	Permitir equipos con el mismo nombre, pero diferente dirección.
Tecnico	id_persona	Clave primaria	Definir completamente a la entidad y evitar repetición de instancias. Corresponde al documento nacional de identidad.
	id_persona	Clave foránea	Desplegar la relación de generalización / especialización con su superclase “Persona”.
	equipo	Clave foránea	Relacionar a “Tecnico” con “Equipo”, a través del atributo (clave primaria para “Equipo”).
Sociedad	id_equipo	Clave primaria	Definir a la entidad y evitar repetición de instancias. Atributo real que se corresponde con el nº de identificación fiscal o similar.
	id_equipo	Clave foránea	Desplegar la relación de generalización / especialización con su superclase “Equipo”.
	tipo_sociedad	Validación del contenido	Asegurar que el tipo de sociedad se encuentre entre los valores tipificados (“SA”, “SAD”, “Sco” o “SL”).
Club	id_equipo	Clave primaria	Definir a la entidad y evitar repetición de instancias. Atributo real que se corresponde con el nº de identificación fiscal o similar.
	id_equipo	Clave foránea	Desplegar la relación de generalización / especialización con su superclase “Equipo”.
Partido	id_partido	Clave primaria	Definir a la entidad y simplificar la clave primaria a un único atributo.
	fecha, hora, pabellon	Unicidad	Evitar la repetición de instancias con diferentes claves primarias, es necesario añadir esta restricción para garantizar la ausencia de repeticiones de instancias.
	municipio	Clave foránea	Relacionar a “Partido” con “Localidad”, a través del atributo (clave primaria para “Localidad”).
	eq_local	Clave foránea	Relacionar a “Partido” con “Equipo”, a través del atributo (clave primaria para “Equipo”).

	eq_visitante	Clave foránea	Relacionar a “Partido” con “Equipo”, a través del atributo (clave primaria para “Equipo”).
	competicion	Clave foránea	Relacionar a “Partido” con “Competicion”, a través del atributo (clave primaria para “Competicion”).
Contrato	id_contrato	Clave primaria	Definir a la entidad y simplificar la clave primaria a un único atributo.
	Jugador, fecha_firma, eq_comprador	Unicidad	Evitar la repetición de instancias con diferentes claves primarias, es necesario añadir esta restricción para garantizar la ausencia de repeticiones de instancias.
	jugador	Clave foránea	Relacionar a “Contrato” con “Jugador”, a través del atributo (clave primaria para “Jugador”).
	eq_comprador	Clave foránea	Relacionar a “Contrato” con “Equipo”, a través del atributo (clave primaria para “Equipo”).
	eq_vendedor	Clave foránea	Relacionar a “Contrato” con “Equipo”, a través del atributo (clave primaria para “Equipo”).
	fecha_ini, fecha_firma	Validación del contenido	Garantizar que la fecha de inicio sea igual o posterior a la fecha de la firma.
	fecha_fin, fecha_ini	Validación del contenido	Garantizar que la fecha de finalización sea igual o posterior a la fecha de inicio.
Eq_arbitral	id_arbitraje	Clave primaria	Definir a la entidad y simplificar la clave primaria a un único atributo.
	partido, arbitro	Unicidad	Evitar la repetición de instancias con diferentes claves primarias, es necesario añadir esta restricción para garantizar la ausencia de repeticiones de instancias.
	partido	Clave foránea	Relacionar a “Eq_arbitral” con “Partido”, a través del atributo (clave primaria para “Partido”).
	arbitro	Clave foránea	Relacionar a “Eq_arbitral” con “Arbitro”, a través del atributo (clave primaria para “Arbitro”).
	rol	Validación del contenido	Asegurar que el rol del árbitro sea un reflejo de la realidad (“principal” o “auxiliar”).
Estadísticas	id_estadistica	Clave primaria	Definir a la entidad y simplificar la clave primaria a un único atributo.
	jugador, partido	Unicidad	Evitar la repetición de instancias con diferentes claves primarias, es necesario añadir esta restricción para garantizar la ausencia de repeticiones de instancias.
	jugador	Clave foránea	Relacionar a “Estadísticas” con “Jugador”, a través del atributo (clave primaria para “Jugador”).
	partido	Clave foránea	Relacionar a “Estadísticas” con “Partido”, a través del atributo (clave primaria para “Partido”).
	min_jug	Validación del contenido	Comprobar que los minutos jugados se hallen dentro del rango establecido por la duración del partido. A fin de simplificar la implementación, se determina que la duración de un partido sea de 40min., sin tener en cuenta prórrogas ni partidos NBA (48 min.).

	t1e, t1i	Validación del contenido	Garantizar que el nº de tiros de 1 punto encestandos sea igual o inferior que el nº de tiros intentados.
	t2e, t2i	Validación del contenido	Garantizar que el nº de tiros de 2 puntos encestandos sea igual o inferior que el nº de tiros intentados.
	t3e, t3i	Validación del contenido	Garantizar que el nº de tiros de 3 puntos encestandos sea igual o inferior que el nº de tiros intentados.
Baja_logica	tabla	Validación del contenido	Asegurar que las bajas insertadas corresponden a tablas existentes en el sistema.

Tabla 8: restricciones de integridad

No se definieron restricciones de integridad para las relaciones del módulo estadístico y la tabla de auditoría, ya que la información a registrar en ambos casos, mediante copia directa o mediante cálculo, provendrá de las relaciones anteriores, que ya han sido controladas.

3.2.3. Decisiones de diseño

A continuación, y para un mejor entendimiento del modelo definido durante la etapa de diseño lógico, se justifican las decisiones de desarrollo que pueden resultar menos evidentes:

- La relación “Baja”, creada a partir de las entidades “Baja” y “Jugador” permite almacenar el histórico de bajas de cada jugador.
- Se crean atributos identificadores para aquellas relaciones con clave primaria múltiple (“Partido”, “Contrato”, “Eq_arbitral”, “Baja”, y “Estadísticas”) que consiguen identificar por completo cada instancia, sin necesidad de recurrir a sus atributos foráneos. De esta manera se simplifica la gestión de las interrelaciones en las que estas relaciones participan, haciendo las claves foráneas más ligeras.
- Los datos relacionados con las estadísticas de los partidos se han implementado por medio de la interrelación entre las entidades “Partido” y “Jugador”. La relación resultante (“Estadísticas”) define también la convocatoria de un jugador a un partido. El número de jugadores convocados para un partido se controlará mediante el uso de disparadores para asegurar que sea el correcto.
- Debido a la existencia de dos configuraciones diferentes de equipos arbitrales, se decide implementar su registro mediante una interrelación entre las entidades “Partido” y “Arbitro”. De este modo se obtiene flexibilidad para albergar equipos arbitrales de cualquier tamaño fácilmente.
- El número de árbitros que componen un equipo arbitral se controlará mediante disparadores. La estructura del equipo arbitral también se verificará para garantizar la existencia de un solo árbitro principal, y dos o tres auxiliares, dependiendo del tipo de competición.
- Se desestimó la creación de una entidad para las estadísticas de cada partido por los siguientes motivos:

- La entidad debería ser débil, ya que necesita de “Partido” y “Jugador” para quedar identificada. De hecho no es posible seleccionar ningún atributo propio que pueda utilizarse como clave primaria, para ello se debería incluir un atributo identificador.
- Por tanto, y al tratarse de una entidad débil, necesitaría participar de una interrelación con una entidad asociativa (“Convocatoria”) creada a partir de la interrelación entre “Partido” y “Jugador”.
- Toda interrelación que incluya a una entidad débil ha de desplegarse mediante una conectividad 1:N, ésta deberá posicionarse en el extremo N.
- La interrelación entre “Convocatoria” y “Estadísticas” resultaría en una conectividad 1:1, ya que cada jugador convocado para un partido genera un único estudio estadístico. Por tanto, dicha solución no es viable.
- La opción de una interrelación ternaria entre “Jugador”, “Partido” y “Estadísticas” no es posible debido a la ausencia de clave primaria en la entidad “Estadísticas”.
- La interrelación “Contrato” se ha desarrollado siguiendo las mismas directrices que “Estadísticas”, ya justificada en el punto anterior.
- Mediante los atributos “fecha-ini” y “fecha-fin” de la relación resultante “Contrato” se permite el registro de contratos de cualquier duración.
- El atributo “rol” de la relación resultante “Eq_Arbitral” informa del papel de cada árbitro en el encuentro. Así sólo serán válidos los valores “principal” y “auxiliar”, hecho que deberá ser controlado para evitar errores.
- La relación “Cuerpo_Técnico” ofrece flexibilidad para que cualquier estructura pueda ser representada.

3.2.4. Validación

Durante la validación se comprobó que el resultado de la fase del diseño lógico cumpliera con los requerimientos aplicables recogidos durante la etapa de definición de requerimientos funcionales, así como su coherencia con el resultado de la etapa anterior.

A continuación se muestra todos los conceptos verificados:

Requerimiento	Resultado
Conceptos incluidos en el diseño conceptual.	El diseño lógico ha integrado todas las entidades e interrelaciones definidas durante el diseño conceptual.
Información mínima requerida.	Se han mantenido todos los atributos directamente relacionados con entidades, y se han

	incluido todos aquellos relacionados con interrelaciones.
Integridad de unicidad de la clave primaria.	Todas las relaciones disponen de una clave primaria que identifica inequívocamente a cada instancia.
Integridad del dominio.	Se ha observado la necesidad de controlar aquellos atributos que disponen de un conjunto limitado de valores válidos, y que no pueden ser verificados únicamente mediante la pertenencia al dominio del atributo.
Normalización a FNBC.	Ninguna relación muestra la existencia de varias claves candidatas compuestas que compartan algún elemento.
Flexibilidad.	Las relaciones derivadas de interrelaciones se han desarrollado contemplando futuras ampliaciones en el sistema. Se ha asegurado que aquellos conceptos que pudieran presentar falta de uniformidad entre instancias presenten flexibilidad suficiente para adaptarse a cualquier caso. Un ejemplo lo constituye la relación ‘Cuerpo_Técnico’.
Eficiencia	Se han simplificado, en lo posible, las claves primarias mediante atributos identificadores, a fin de mejorar el rendimiento de la base de datos. Ya que de este modo se consiguen búsquedas más rápidas.

Tabla 9: validación del diseño lógico

3.3. Diseño físico

La fase de diseño físico tuvo como objetivo determinar el formato que adoptaría el sistema en el/los servidores que lo acogieran, a partir de la estructura obtenida al final del diseño lógico, de manera que el rendimiento fuese acorde con las necesidades del cliente.

3.3.1. Decisiones de diseño

Considerando que la base de datos, debido al concepto al que está relacionada y al número de potenciales usuarios, no se hallará sujeta a una alta demanda, y con el objetivo de simplificar su diseño físico dentro de las posibilidades del equipo desarrollador y del sistema gestor a utilizar (Oracle XE), se estableció su implementación sobre un único servidor, y no en una configuración en malla.

Dado que el rendimiento del sistema fue un objetivo prioritario durante la fase del diseño físico, todos los procedimientos almacenados contemplaron la gestión de transacciones, a fin de evitar la ocupación innecesaria de la base de datos.

Con el objetivo de agilizar las consultas, se crearon índices para aquellas tablas con mayor volumen de datos almacenados y, en concreto, para aquellos atributos, claves foráneas, con mayor demanda por parte de procedimientos. Inicialmente fueron las tablas que contenían las relaciones “Estadísticas”, “Partido” y “Contrato” las que dispongan de índices, aunque según el volumen de datos aumente, o las necesidades de información varíen, podrían añadirse nuevos índices. Aunque el sistema crea automáticamente índices para todas las claves primarias y atributos con restricciones de valor único, claves primarias y valores únicos, se efectuó su traspaso al espacio de tabla definido para los índices.

Índice	Tabla	Atributo	Motivo
I_Partido_competicion	Partido	competicion	Clave foránea de alta demanda en consultas.
I_Partido_eq_local	Partido	eq_local	Clave foránea de alta demanda en consultas.
I_Partido_eq_visitante	Partido	eq_visitante	Clave foránea de alta demanda en consultas.
I_Contrato_id_contrato	Contrato	eq_comprador	Clave foránea de alta demanda en consultas.
I_Contrato_jugador	Contrato	jugador	Clave foránea de alta demanda en consultas.
I_Estadisticas_jugador	Estadisticas	jugador	Clave foránea de alta demanda en consultas.
I_Estadisticas_partido	Estadisticas	partido	Clave foránea de alta demanda en consultas.

Tabla 10: índices creados

Los objetos de la base de datos se distribuyeron entre 3 espacios de tabla:

- **DATOS:** almacena tablas del modelo de negocio, disparadores y procedimientos almacenados.
- **INDICES:** contiene los índices creados.
- **DW_LOG:** guarda las tablas relacionadas con el módulo estadístico, la tabla de bajas lógicas y la tabla de auditoría (*log*). Este espacio se establece con el objetivo de salvaguardar la integridad y veracidad de los datos estadísticos y del registro de llamadas a procedimientos almacenados.

La configuración creada a partir de los parámetros iniciales constituyó el punto de partida del sistema. Una vez en funcionamiento, se podrá ajustar para adecuar su rendimiento a las necesidades reales de uso.

Se crea el usuario con rol administrador con sus propios privilegios para permitir el acceso a los recursos requeridos para la implementación del sistema.

3.3.2. Validación

Durante la validación se comprobó que el resultado de la fase del diseño físico cumpliera con los requerimientos aplicables recogidos durante la etapa de definición de requerimientos funcionales, así como su coherencia con el resultado de la etapa anterior.

A continuación se muestra todos los conceptos verificados:

Requerimiento	Resultado
Conceptos incluidos en el diseño lógico.	El diseño físico, especialmente de los espacios de tabla, se ha diseñado de acuerdo con los requerimientos, tanto funcionales como no funcionales. También se ha contemplado el resultado del diseño lógico, ya que se ha comprobado que los objetos a ubicar en espacios de tablas diferentes no se hallen interrelacionados.
Información mínima requerida.	Se han recogido todas las relaciones resultantes de la fase anterior a la hora de estructurar la base de datos.
Flexibilidad.	El resultado del diseño físico permite realizar futuros cambios sobre el sistema en caso de necesidad.
Eficiencia	La estructura resultante, ha sido creada bajo la premisa de favorecer el rendimiento de la base de datos.
Privilegios	Los permisos de acceso se han creado en alineación con los requerimientos del cliente.

Tabla 11: validación diseño físico

4. IMPLEMENTACIÓN

A partir de los resultados obtenidos durante la fase de diseño, se construyó la base de datos adaptando el modelo a las particularidades del sistema gestor, Oracle 11g XE en este caso.

La fase de implementación se dividió, en lo posible, entre la construcción de los elementos propios del modelo de negocio, y aquellos que constituyen el módulo estadístico y la tabla de auditoría.

4.1. Espacios de tabla

Se implementaron los 3 espacios de tablas determinados durante la fase del diseño físico. La implementación se realizó mediante el script “creación_índices”.

Espacio de tabla	Contenido
DATOS	Tablas del modelo de negocio.
INDICES	Índices de la base de datos, tanto los creados automáticamente, relacionados con las PK y unicidades, como aquellos relativos a FK que se han considerado importantes.
DW_LOG	Tablas del módulo estadístico, de bajas lógicas y log de control de uso de procedimientos almacenados. Este espacio de tabla se crea como sólo lectura. Los procedimientos que se encarguen de actualizar el contenido de las tablas cambiarán esta condición temporalmente, hasta que la actualización esté completa.

Tabla 12: espacios de tabla

La validación de los espacios de tabla se realizó mediante la comprobación de su existencia tras su implementación, y a través del alojamiento de los elementos a contener (a comprobar durante las siguientes fases). El resultado fue satisfactorio en cualquier caso, los espacios de tabla se crearon correctamente y los elementos (tablas e índices) se albergaron correctamente.

4.2. Usuarios

Con el fin de establecer los permisos adecuados para el rol de administrador, necesario para la implementación de la base de datos, se creó el usuario correspondiente al rol de administrador. La implementación de usuarios se efectuó mediante el script “creación_usuarios”.

La validación de los usuarios creados se realizó en función de la accesibilidad permitida. En el caso del usuario administrador, se obtuvieron los permisos requeridos.

4.3. Tablas

La implementación de las tablas incluyó todas las tablas del sistema, tanto las del modelo de negocio, como las del módulo estadístico, como la tabla de auditoría. La construcción se llevó a cabo mediante el script “creación_tablas”.

La validación de las tablas se realizó en dos etapas, en una primera se comprobó que las tablas se creasen sin errores, y en una segunda se aplicó un exhaustivo juego de pruebas que verificó todas las restricciones de integridad incluidas en cada tabla. Para las restricciones más generales, como valores nulos, duplicación de claves primarias o tipos de datos incorrectos, se ejecutó una única prueba.

4.4. Índices

Se crearon, explícitamente, índices para aquellas claves foráneas ubicadas en tablas de alto volumen o alta demanda, con el fin de facilitar las consultas sobre ellas. El script “creación_índices” se encarga de reubicar los índices creados por el sistema, relacionados con claves primarias y unicidades de todas las tablas, y de crear los nuevos índices.

Índice	Tabla	Atributo	Restricción de integridad	Tabla referenciada
I_Partido_competicion	Partido	competicion	Clave foránea	Competicion
I_partido_eq_local	Partido	eq_local	Clave foránea	Equipo
I_partido_eq_visitante	Partido	eq_visitante	Clave foránea	Equipo
I_Contrato_eq_comprador	Contrato	eq_comprador	Clave foránea	Equipo
I_Contrato_jugador	Contrato	jugador	Clave foránea	Jugador
I_Estadisticas_jugador	Estadisticas	jugador	Clave foránea	Jugador
I_Estadisticas_partido	Estadisticas	partido	Clave foránea	Partido

Tabla 13: índices creados

4.5. Secuencias

Se crearon secuencias para facilitar la asignación automática a aquellos atributos creados para simplificar las claves primarias multiatributo de aquellas tablas que no disponían de una clave primaria natural de un único atributo. A fin de evitar repeticiones en los valores de las diferentes secuencias, situación que podría dificultar la comprensión del sistema, se definieron diferentes rangos como dominio de cada atributo.

Secuencia	Atributo	Tabla	Rango
sq_id_localidad	id_localidad	Localidad	1000000 - 1999999

sq_id_competicion	id_competicion	Competicion	2000000 - 2999999
sq_id_partido	id_partido	Partido	3000000 - 3999999
sq_id_contrato	id_contrato	Contrato	4000000 - 4999999
sq_id_estadistica	id_estadistica	Estadisticas	5000000 - 5999999
sq_id_arbitraje	id_arbitraje	Eq_arbitral	6000000 - 6999999
sq_id_log	id_log	Log_procs	7000000 - 7999999

Tabla 14: secuencias

4.6. Disparadores

Mediante la creación de disparadores asociados a las secuencias definidas en el punto anterior, se posibilitó que cada inserción realizada sobre cada una de las tablas afectadas (“Localidad”, “Competicion”, “Partido”, “Contrato”, “Estadisticas”, “Eq_arbitral” y “Log_procs”) activara la generación de un nuevo valor, dentro del rango establecido y consecutivo al último existente.

Se validó el funcionamiento conjunto de disparadores y secuencias, creando nuevas instancias de las tablas afectadas. Las pruebas consistieron en insertar nuevas instancias con un valor aleatorio o un valor “*null*” para el atributo autonumérico. Todas las pruebas resultaron satisfactorias. No fue posible realizar la prueba omitiendo el valor relacionado con el atributo autonumérico, ya que en todos los casos ejercen la función de clave primaria. Así fue necesario insertar un valor aleatorio para crear una nueva instancia; el disparador ya se encarga de asignar el valor correcto al atributo. Se comprobó también el comportamiento del sistema en caso de borrado de instancias. Se confirmó que la secuencia continúa normalmente y no se reutilizan los valores eliminados.

Se aprovechó el mismo juego de pruebas utilizado para la validación de las tablas para la inserción de valores.

4.7. Procedimientos ABM

Los procedimientos ABM tienen por objetivo gestionar las operaciones de alta, baja y modificación en las tablas implementadas en el sistema. De esta manera se garantiza la correcta gestión de errores y el registro en la tabla de auditoría de las diferentes operaciones realizadas.

Tal y como indican las especificaciones de cliente, y con la finalidad de optimizar los recursos de tiempo disponibles, se implementaron procedimientos ABM únicamente sobre las tablas de mayor relevancia (“Jugador”, “Contrato” y “Partido”).

4.7.1. IMPLEMENTACIÓN

A continuación se detallan los diferentes procedimientos ABM implementados:

Procedimiento	Descripción	Excepciones gestionadas
Proc_alta_contrato	Incorpora un nuevo registro a la tabla "Contrato". Comprueba si el jugador ya tiene contrato en vigencia en las fechas del contrato propuesto. Comprueba si la persona relacionada, el jugador o el equipo están dados de baja del sistema. Asegura la herencia total de los equipos (club o sociedad). El procedimiento retorna "OK" si acaba con éxito, o "Error" + explicación en caso contrario.	-Jugador con contrato ya vigente. -Jugador inactivo. -Jugador en baja lógica. -Persona relacionada con el jugador en baja lógica. -Equipo en baja lógica. -Equipos definidos como club o sociedad (herencia total).
Proc_alta_jugador	Incorpora un nuevo registro a la tabla "Jugador". Comprueba si persona o representante (incluyendo persona de representante) están dados de baja lógica. Comprueba si la persona tiene otro rol en el sistema (herencia disjunta). El procedimiento retorna "OK" si acaba con éxito, o "Error" + explicación en caso contrario.	-Persona ya ejerce otro rol en el sistema. -Persona relacionada con el jugador en baja lógica. -Representante en baja lógica *. <i>*La persona relacionada con el representante se controla durante el alta de éste..</i>
Proc_alta_partido	Incorpora un nuevo registro a la tabla "Partido". Comprueba que el municipio, la competición o alguno de los equipos no estén dados de baja en el sistema. Asegura la herencia total de los equipos (club o sociedad) y de la competición (nacional o continental). El procedimiento retorna "OK" si acaba con éxito, o "Error" + explicación en caso contrario.	-Municipio en baja lógica. -Equipo en baja lógica. -Competición en baja lógica. -Equipos definidos como club o sociedad (herencia total). -Competición definida como nacional o continental (herencia total).
Proc_baja_contrato	Incorpora un nuevo registro a la tabla "Baja_lógica". Comprueba si el contrato a dar de baja existe en el sistema. El procedimiento retorna "OK" si acaba con éxito, o "Error" + explicación en caso contrario.	-El elemento no se encuentra en la tabla de bajas lógicas.
Proc_baja_jugador	Incorpora un nuevo registro a la tabla "Baja_lógica". Comprueba si el jugador a dar de baja existe en el sistema. El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.	-El elemento no se encuentra en la tabla de bajas lógicas
Proc_baja_partido	Incorpora un nuevo registro a la tabla "Baja_lógica". Comprueba si el partido a dar de baja existe en el sistema. El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.	-El elemento no se encuentra en la tabla de bajas lógicas

<p>Proc_modif_contrato</p>	<p>Modifica un registro existente en la tabla "Contrato". Permite la modificación de cualquier atributo, excepto el identificador del contrato (PK). Comprueba que no exista solape con contrato existente en caso modificación fecha inicio. Comprueba persona, jugador o equipo no estén dados de baja en el sistema. Asegura la herencia total de los equipos (club o sociedad). El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.</p>	<p>-Jugador ya dispone de otro contrato vigente en las fechas definidas. -Jugador inactivo. -Jugador en baja lógica (caso modificación atributo "jugador"). -Persona relacionada con el jugador en baja lógica (caso modificación atributo "jugador"). -Equipo en baja lógica (caso modificación atributo "equipo"). -Equipos definidos como club o sociedad (herencia total).</p>
<p>Proc_modif_jugador</p>	<p>Modifica un registro existente en la tabla "Jugador". Permite la modificación de cualquier atributo, excepto el identificador del jugador (PK). Comprueba que persona o representante no ejerzan otro rol en el sistema (herencia disjunta). Comprueba que persona, jugador o representante no estén dados de baja en el sistema. El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.</p>	<p>-Jugador en baja lógica (caso modificación atributo "jugador"). -Persona relacionada con el jugador en baja lógica (caso modificación atributo "jugador"). -Representante ya ejerce otro rol (caso modificación atributo "representante"). -Representante en baja lógica* (caso modificación atributo "representante"). <i>*La persona relacionada con el representante se controla durante el alta de éste..</i></p>
<p>Proc_modif_partido</p>	<p>Modifica un registro existente en la tabla "Partido". Permite la modificación de cualquier atributo, excepto el identificador del partido (PK). Comprueba que municipio, equipo o competición no estén dados de baja en el sistema. Asegura la herencia total de los equipos (club o sociedad) y de la competición (nacional o continental). El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.</p>	<p>-Municipio en baja lógica (caso modificación atributo "municipio"). -Equipo en baja lógica (caso modificación atributos "eq_local" o "eq_visitante"). -Competición en baja lógica (caso modificación atributo "competicion"). -Equipos definidos como club o sociedad (herencia total). -Competición definida como nacional o continental (herencia total).</p>

Tabla 15: procedimientos ABM

4.7.2. VALIDACIÓN

La validación de los procedimientos para casos de finalización correcta se realizó mediante el llenado de las tablas de la base de datos. Se verificó que todas las inserciones, modificaciones y bajas con parámetros de entrada correctos se realizaran correctamente, comprobando el contenido de la tabla log .

Para la verificación de parámetros incorrectos se habilitó un juego de pruebas, “pruebas_procs_y_disp_her” incluida en la carpeta “SCRIPTS”, que provocaba todos los errores gestionados por los procedimientos y los registraba en el archivo “resultados_pruebas_procs_ABM”, incluido en la carpeta “RESULTADOS_PRUEBAS”. Los resultados también pueden ser comprobados en la tabla de auditoría.

4.8. Procedimientos de relaciones especiales

Dado su especial dificultad, se implementan procedimientos de alta para las relaciones “Estadísticas” y “Eq_arbitral”.

4.8.1. IMPLEMENTACIÓN

Los procedimientos “Proc_alta_estadísticas” y “Proc_alta_eq_arb” permiten el control del número de elementos necesarios para conformar las convocatorias de los equipos (12 jugadores por equipo) en el caso de las estadísticas, y el número de árbitros, en función de la competición, en el caso de los equipos arbitrales. Mediante la actualización de los atributos “est_convocatoria” y “est_eq_arb” de la relación “Partido”, los procedimientos actualizan el estatus de las convocatorias, estadísticas y equipos arbitrales.

El atributo “est_convocatoria” mostrará “OK” si existen 12 + 12 estadísticas registradas, y el tiempo total jugado sea de 200 + 200 minutos. En caso de no estar completo indicará el número de jugadores pendientes de introducir de cada equipo.

El atributo “est_eq_arb” mostrará “OK” en el caso de que exista un árbitro principal y varios auxiliares, determinado por la competición a la que pertenezca el partido. En caso de no estar completo indicará el número de árbitros de cada tipo pendientes de introducir.

El procedimiento de alta de estadísticas contempla que , ya que en la realidad las convocatorias no se conocen hasta minutos antes del partido, la introducción de los valores de convocatoria y estadísticas se realicen conjuntamente después del partido. Así pues, se desestima, de entrada, el registro de convocatoria de manera aislada, por no tener aplicación práctica y complicar la gestión de estadísticas innecesariamente. El contenido del atributo “est_convocatoria” será requerido por diferentes consultas y por el módulo estadístico, y la existencia de diversos estatus (convocatoria completa o incompleta, estadísticas completas o incompletas) dificultaría su gestión.

Procedimiento	Descripción	Excepciones gestionadas
Proc_alta_estadísticas	<p>Incorpora un nuevo registro a la tabla "Estadísticas".</p> <p>Comprueba que la convocatoria del partido esté completa, si ya lo estaba devuelve error y no permite la inserción.</p> <p>En caso de que falten jugadores para completar la convocatoria relacionada con el partido, actualiza el campo "estatus" de la tabla "Partido".</p> <p>Comprueba que la convocatoria del equipo esté completa.</p> <p>Comprueba que la convocatoria del partido esté completa.</p> <p>Comprueba que el tiempo jugado por equipo no sea superior a 200min (40min x 5 jugadores).</p> <p>Comprueba que el tiempo de juego no se complete con menos de 5 jugadores.</p> <p>Comprueba jugador no esté de baja laboral durante disputa partido.</p> <p>Comprueba jugador no estuviese dado de baja lógica durante partido (baja lógica anterior a fecha partido).</p> <p>El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.</p>	<p>-Jugador sin contrato vigente.</p> <p>-Jugador inactivo.</p> <p>-Equipo del jugador no participa en el partido.</p> <p>-Convocatoria del partido ya completa (existen 12 + 12 jugadores registrados para el partido).</p> <p>-Convocatoria del equipo del jugador ya completa (existen 12 jugadores del equipo registrados para el partido).</p> <p>-Tiempo de juego insuficiente para el equipo (caso de completar la convocatoria de 12 jugadores sin llegar a los 200 minutos de juego).</p> <p>-Exceso de tiempo de juego (caso de que la suma de tiempo jugado por todos los jugadores insertados del equipo supere los 200 minutos).</p> <p>-Jugador en baja lógica.</p> <p>-Persona relacionada con jugador en baja lógica.</p> <p>-Jugador en baja laboral durante la disputa del partido.</p>
Proc_alta_eq_arb	<p>Comprueba que el estado del equipo arbitral de un partido.</p> <p>En caso de que falten árbitros para completar el equipo arbitral relacionado con el partido, actualiza el campo "estatus" de la tabla "Partido". Si ya está completo, devuelve error y evita la inserción.</p> <p>Comprueba que no se supere el nº de árbitros principales y auxiliares, en función de la competición.</p> <p>Comprueba que el árbitro y el partido no se hallen dados de baja del sistema.</p> <p>El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.</p>	<p>-Equipo arbitral ya completo.</p> <p>-Árbitro principal ya insertado (en caso de introducción de un árbitro principal).</p> <p>-Número de árbitros auxiliares ya completo (en caso de introducción de un árbitro auxiliar).</p> <p>-Árbitro en baja lógica.</p> <p>-Persona relacionada con árbitro en baja lógica.</p> <p>-Partido en baja lógica.</p>

Tabla 16: procedimientos de relaciones críticas

4.8.2. VALIDACIÓN

La validación de los procedimientos para casos de finalización correcta se realiza mediante el llenado de las tablas de la base de datos. Se comprueba que todas las inserciones, con parámetros de entrada correctos se realicen correctamente, comprobando el contenido de la tabla log (se adjunta archivo “resultados_introduccion_inicial” en la carpeta “RESULTADOS_PRUEBAS”).

Para la verificación de parámetros incorrectos se habilita un juego de pruebas, “pruebas_procs_y_disp_her” incluida en la carpeta “SCRIPTS”, que provoca todos los errores gestionados por los procedimientos y registra los errores en el archivo “resultados_pruebas_procs”, incluido en la carpeta “RESULTADOS_PRUEBAS”. Los resultados también pueden ser comprobados en la tabla de auditoría.

4.9. Disparadores de herencia disjunta

Las relaciones de herencia definidas durante la fase de diseño establecían las siguientes herencias:

Superclase	Subclases	Tipo de herencia
Persona	Tecnico, Jugador, Repr_jug y Arbitro	Disjunta y parcial
Equipo	Sociedad y Club	Disjunta y total
Competicion	Nacional y Continental	Disjunta y total

Tabla 17: herencias

La cualidad de herencia disjunta (una instancia sólo puede pertenecer a una subclase) se ha garantizado mediante los procedimientos ABM, o a través del uso de disparadores para el resto de relaciones.

La característica de herencia total (todas las instancias deben pertenecer a alguna de las subclases) o parcial (las instancias pueden no pertenecer a ninguna de las subclases) se ha controlado exclusivamente mediante procedimientos ABM, ya que sólo afecta a las relaciones “Contrato” y “Partido”, en las que se gestionan equipos y competiciones, que son las únicas relaciones definidas como herencias totales.

4.9.1. IMPLEMENTACIÓN

A continuación se detallan los disparadores creados con la finalidad de controlar la herencia disjunta para aquellas relaciones que no disponen de procedimientos ABM:

Disparador	Regla de activación	Superclase
tr_hd_tecnico	Antes de inserción o modificación de la tabla "Técnico"	Persona
tr_hd_repr	Antes de inserción o modificación de la tabla "Repr_jug"	
tr_hd_arb	Antes de inserción o modificación de la tabla "Arbitro"	
tr_hd_soc	Antes de inserción o modificación de la tabla "Sociedad"	Equipo
tr_hd_club	Antes de inserción o modificación de la tabla "Club"	
tr_hd_nac	Antes de inserción o modificación de la tabla "Nacional"	Competicion
tr_hd_cont	Antes de inserción o modificación de la tabla "Continental"	

Tabla 18: disparadores control herencia disjunta

Los disparadores devuelven un mensaje de error en caso de que se incumpla la regla de herencia disjunta.

Se adjunta script de creación de los disparadores en el archivo "creación_disparadores" incluido en la carpeta "SCRIPTS".

4.9.2. VALIDACIÓN

Para la verificación del funcionamiento de los disparadores se habilitó un juego de pruebas, "pruebas_procs_y_disp_her" incluida en la carpeta "SCRIPTS", que provocaba todos los errores gestionados por los procedimientos y efectuaba su registro en el archivo "resultados_pruebas_disp_her_disj", incluido en la carpeta "RESULTADOS_PRUEBAS".

4.10. Consultas

De acuerdo con los requerimientos del cliente, se implementan una serie de consultas estándar que permiten acceder a la información contenida en el sistema, tras cumplir unas condiciones de búsqueda y, en algunos casos, tras realizar algunos cálculos.

4.10.1. IMPLEMENTACIÓN

La implementación de las consultas especificadas se llevó a cabo mediante procedimientos almacenados. Como en el resto de procedimientos ya implementados se gestionan las excepciones, y las llamadas se registran en la tabla de auditoría (“Log_procs”).

A continuación se detallan los diferentes procedimientos de consulta implementados.

Procedimiento	Descripción	Excepciones gestionadas
Consulta_A	<ul style="list-style-type: none"> -Devuelve el listado de los jugadores de una competición especificada como parámetro de entrada. -Comprueba que los elementos relacionados con la consulta no se hallen en baja lógica. -Comprueba que la competición exista. -El procedimiento devuelve el resultado de la consulta por pantalla. -El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario. 	<ul style="list-style-type: none"> -Competición en baja lógica. -Competición no existente.
Consulta_B	<ul style="list-style-type: none"> -Devuelve el listado de equipos de una competición, en su última edición, ordenados por el nº de puntos a favor (descendente). -Devuelve nombre del equipo, nº de puntos a favor y año de la competición, como comprobación de que se trata de la última edición. -Comprueba que los elementos relacionados con la consulta no se hallen en baja lógica. -Comprueba que la competición exista. -El procedimiento devuelve el resultado de la consulta por pantalla. -El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario. 	<ul style="list-style-type: none"> -Competición en baja lógica. -Competición no existente.

<p>Consulta_C</p>	<ul style="list-style-type: none"> -Devuelve el listado de los 5 mejores jugadores por cada posición, en función de un ámbito de competición (especificado por el país o continente en el que se han desarrollado los campeonatos a explorar) , un género y una modalidad (parámetros de entrada). -En caso de existir menos de 5 jugadores para la posición, sólo mostrará los disponibles. -Sólo se tienen en cuenta aquellas estadísticas en las que la valoración ponderada disponga de valor (no null), con el fin de evitar errores y no perjudicar la valoración global del jugador. -Devuelve el identificador, el nombre, la posición, la media de la valoración ponderada de cada jugador y la posición de éste dentro del ranking por posición. -Al tratarse de una consulta del histórico de estadísticas, no se tiene en cuenta si el jugador se halla en baja lógica o inactivo, ya que el sistema ya regula que no se introduzcan estadísticas de jugadores en baja lógica. -Sí se comprueban que el resto de elementos relacionados con la consulta no se hallen en baja lógica. -Comprueba que el ámbito (país o continente) exista. -Comprueba que el género y la modalidad sean correctos (de acuerdo con las restricciones de integridad definidas en la tabla "Jugador"). -El procedimiento devuelve el resultado de la consulta por pantalla. -El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario. 	<ul style="list-style-type: none"> -Ambito inexistente. -Género incorrecto. -Modalidad incorrecta.
-------------------	--	---

Consulta_D	<p>-Devuelve el nº de contratos firmados por un representante determinado, durante un año concreto, así como el listado del valor total de la operación para cada uno de ellos y su identificador para comprobar que no existen repeticiones.</p> <p>-El año y el identificador del representante conforman los parámetros de entrada.</p> <p>-Comprueba que ni el representante, ni la persona relacionada, ni el contrato se hallen en baja lógica (los contratos dados de baja no se contabilizarán).</p> <p>-El procedimiento devuelve el resultado de la consulta por pantalla.</p> <p>-El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.</p>	<p>-Representante en baja lógica.</p> <p>-Persona relacionada con representante en baja lógica.</p>
Consulta_E	<p>-Devuelve el listado de los 10 equipos que más dinero han gastado en fichajes durante un año concreto, proporcionado como parámetro de entrada, ordenados de manera descendente.</p> <p>-Devuelve nombre del equipo, el dinero gastado. Comprueba si durante el año se han firmado contratos, en caso de que no existan contratos firmados durante el año, devuelve error.</p> <p>-El procedimiento no tiene en cuenta equipos ni contratos en baja lógica.</p> <p>-El procedimiento devuelve el resultado de la consulta por pantalla.</p> <p>-El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.</p>	<p>-Año sin contratos firmados.</p>
Consulta_F	<p>-Devuelve el listado de los jugadores que acaban contrato al final de la presente temporada, o que están en activo pero sin contrato, en función de un país, un género y una modalidad (parámetros de entrada).</p> <p>-La consulta no tiene en cuenta aquellos jugadores en baja lógica.</p> <p>-Devuelve el identificador y el nombre de cada jugador.</p> <p>-Comprueba que el país exista.</p> <p>-Comprueba que el género y la modalidad sean correctos (de acuerdo con las restricciones de integridad definidas en la tabla "Jugador").</p> <p>-El procedimiento devuelve el resultado de la consulta por pantalla.</p> <p>-El procedimiento registra en la tabla log "OK" si acaba con éxito, o "Error" + explicación en caso contrario.</p>	<p>-País inexistente.</p> <p>-Género incorrecto.</p> <p>-Modalidad incorrecta.</p>

Tabla 19: procedimientos de consulta

4.10.2. VALIDACIÓN

Para la verificación del funcionamiento de los procedimientos de consulta se habilitó un juego de pruebas, “pruebas_consultas” incluida en la carpeta “SCRIPTS”. Las pruebas consistieron en la ejecución de una consulta posible, de acuerdo a los datos iniciales de la base de datos, y en la provocación de todos los errores gestionados, para cada uno de los procedimientos.

Los resultados de las consultas correctas fueron contrastados mediante la verificación, por muestreo, de algunos valores de salida que fueron recalculados, o reconsultados, manualmente. El funcionamiento de la gestión de excepciones se testeó mediante la comparación de la salida obtenida y la esperada, de acuerdo a la gestión de errores definida en el procedimiento.

Los resultados de la ejecución del juego de pruebas se hallan en el archivo “resultados_pruebas_consultas”, incluido en la carpeta “RESULTADOS_PRUEBAS”.

4.11. Módulo estadístico

El módulo estadístico pretende poner a disposición del usuario información tratada (indicadores) relacionados con la población de datos disponible en el sistema.

Entre los requerimientos del cliente se hallan 7 indicadores diferentes que han de proporcionar respuesta inmediata, mediante consulta directa, a los diferentes usuarios. La información ha de hallarse disponible y actualizada en cualquier momento para su consulta.

4.11.1. IMPLEMENTACIÓN

La implementación del módulo estadístico contempló dos elementos diferenciados, por una parte los almacenes de información (alojan la información tratada y actualizada de un indicador para su consulta) y por otra los procesos de extracción, transformación y carga (ETL), que se encargan de actualizar los datos contenidos en los almacenes en función de los cambios sufridos por las tablas de origen del modelo de negocio.

Para los almacenes de datos se crearon tablas que contienen los campos necesarios para dar respuesta a la necesidad de información creada por cada indicador. Estas tablas contendrán únicamente los registros necesarios resultantes del tratamiento de datos aplicado a las tablas origen del modelo de negocio. El detalle de las tablas correspondientes al módulo estadístico puede encontrarse en el punto 3.1.2 de este documento.

En cuanto a los procesos ETL, se ha optado por el uso de disparadores que actúan ante eventos de inserción, actualización o borrado en las tablas origen del modelo de negocio que proporcionan la información principal para el cálculo de los indicadores.

A continuación se detalla la implementación de las diferentes consultas estadísticas.

Tabla	Disparador	Descripción del disparador
DW_activos	tr_dw_jug_activos	<ul style="list-style-type: none"> -Evento: antes de inserción, modificación o borrado en la tabla “Jugador”. -Granularidad: para cada línea afectada. -Actualiza (sumando o restando, según el evento) la tabla DW_activos, si ya se halla inicializada, sólo actualiza la combinación género/modalidad afectada por el cambio. -Si es la primera entrada para la combinación, crea la entrada en la tabla, con número de jugadores activos=1. Los procedimientos ABM se encargan de asegurar que no se hace uso de un jugador o persona en baja lógica.
	tr_dw_jug_activos_tabla_vacia	<ul style="list-style-type: none"> -Evento: antes de inserción en la tabla “Jugador”. -Granularidad: una vez por sentencia. -Inicializa la tabla DW_activos si ésta está vacía y existen datos en el sistema (caso reinicialización del módulo estadístico). El disparador tr_dw_jug_activos se encarga de actualizar con los datos del nuevo registro.
	tr_dw_baja_activos	<ul style="list-style-type: none"> -Evento: antes de inserción, modificación o borrado en la tabla “Baja_logica”. -Granularidad: para cada línea afectada. -Se encarga de actualizar (sumando o restando, según el evento) sólo el registro afectado (combinación género/modalidad) por la modificación de la tabla de bajas lógicas.
DW_annotador	tr_dw_annotador	<ul style="list-style-type: none"> -Evento: después de inserción, modificación o borrado en la tabla “Estadísticas”. -Granularidad: una vez por sentencia. -Disparador que actualiza la tabla DW_annotador con las competiciones en su última temporada. -En caso de existir empate, el sistema guarda todos los máximos anotadores para la competición. -Se comprueban los elementos relacionados con el indicador estadístico(persona, jugador, partido, competición y estadísticas) para no incorporar ninguno en baja lógica.
	tr_dw_annotador_bajas	<ul style="list-style-type: none"> -Evento: después de inserción, modificación o borrado en la tabla “Baja_logica”. -Granularidad: una vez por sentencia. -Disparador que actualiza la tabla DW_annotador con las competiciones en su última temporada. -En caso de existir empate, el sistema guarda todos los máximos anotadores para la competición. -Se comprueban los elementos relacionados con el indicador estadístico(persona, jugador, partido, competición y estadísticas) para no incorporar ninguno en baja lógica.

DW_salario_max	tr_dw_salario	-Evento: después de inserción, modificación o borrado en la tabla “Contrato”. -Granularidad: una vez por sentencia. -Disparador que actualiza la tabla DW_salario_max con los jugadores mejor pagados para cada combinación género/modalidad. -En caso de existir empate, el sistema guarda todos los máximos salarios por género/modalidad. -El jugador debe estar activo para ser tenido en cuenta. -Se comprueban los elementos relacionados con el indicador estadístico(persona, jugador, y contrato) para no incorporar ninguno en baja lógica.
	tr_dw_salario_bajas	-Evento: después de inserción, modificación o borrado en la tabla “Baja_logica”. -Granularidad: una vez por sentencia. -Disparador que actualiza la tabla DW_salario_max con los jugadores mejor pagados para cada combinación género/modalidad. -En caso de existir empate, el sistema guarda todos los máximos salarios por género/modalidad. -El jugador debe estar activo para ser tenido en cuenta. -Se comprueban los elementos relacionados con el indicador estadístico(persona, jugador, y contrato) para no incorporar ninguno en baja lógica.
DW_fortuna	tr_dw_fortuna	-Evento: después de inserción, modificación o borrado en la tabla “Contrato”. -Granularidad: una vez por sentencia. -Disparador que actualiza la tabla DW_fortuna con el jugador con mayores ingresos a lo largo de su carrera deportiva. -En caso de existir empate, el sistema guarda todas las máximas fortunas. -Al tener un carácter histórico, se tienen en cuenta también aquellos contratos de jugadores que en la actualidad se hallen inactivos. -Se comprueban los elementos relacionados con el indicador estadístico (persona, jugador y contrato) para no incorporar ninguno en baja lógica.
	tr_dw_fortuna_bajas	-Evento: después de inserción, modificación o borrado en la tabla “Baja_logica”. -Granularidad: una vez por sentencia. -Disparador que actualiza la tabla DW_fortuna con el jugador con mayores ingresos a lo largo de su carrera deportiva. -En caso de existir empate, el sistema guarda todas las máximas fortunas. -Al tener un carácter histórico, se tienen en cuenta también aquellos contratos de jugadores que en la actualidad se hallen inactivos. -Se comprueban los elementos relacionados con el indicador estadístico (persona, jugador y contrato) para no incorporar ninguno en baja lógica.

DW_sueldo_medio	tr_dw_sueldo_medio_ctr	<p>-Evento: después de inserción, modificación o borrado en la tabla “Contrato”.</p> <p>-Granularidad: una vez por sentencia.</p> <p>-Disparador que actualiza la tabla DW_sueldo medio con el salario medio de los jugadores de cada combinación país, temporada, género y modalidad.</p> <p>-Al tener un carácter histórico, caso de temporadas anteriores, no se tiene en cuenta si el jugador se halla activo o inactivo.</p> <p>-Se comprueban los elementos relacionados con el indicador estadístico (competición, competición nacional, partido y contrato) para no incorporar ninguno en baja lógica.</p>
	tr_dw_sueldo_medio_bajas	<p>-Evento: después de inserción, modificación o borrado en la tabla “Baja_logica”.</p> <p>-Granularidad: una vez por sentencia.</p> <p>-Disparador que actualiza la tabla DW_sueldo medio con el salario medio de los jugadores de cada combinación país, temporada, género y modalidad.</p> <p>-Al tener un carácter histórico, caso de temporadas anteriores, no se tiene en cuenta si el jugador se halla activo o inactivo.</p> <p>-Se comprueban los elementos relacionados con el indicador estadístico (competición, competición nacional, partido y contrato) para no incorporar ninguno en baja lógica.</p>
DW_mejores_equipos	tr_dw_mejores_est	<p>-Evento: después de inserción, modificación o borrado en la tabla “Estadísticas”.</p> <p>-Granularidad: una vez por sentencia.</p> <p>-Disparador que actualiza la tabla DW_mejores_equipos con el top 3 de los mejores equipos ofensivos (mayor nº de puntos marcados=) y el top 3 de los mejores equipos defensivos (menor nº de puntos encajados) para cada combinación de competición y temporada.</p> <p>-En caso de empate, el sistema mostrará todos los equipos con el mismo valor de puntos que encajen dentro del correspondiente top 3.</p> <p>-Al tener un carácter histórico, caso de temporadas anteriores, no se tiene en cuenta si el jugador se halla activo o inactivo.</p> <p>-Los top 3 de equipos ofensivos muestran la totalidad de puntos anotados en negativo, mientras que los top 3 de equipos defensivos lo hacen en negativo diferencia entre puntos totales anotados en los partidos en los que el equipo ha participado y los puntos anotados por el equipo en cuestión).</p> <p>-Se comprueban los elementos relacionados con el indicador estadístico (competición, estadística, equipo, partido y contrato) para no incorporar ninguno en baja lógica.</p>

	Tr_dw_bajas _mejores	<p>-Evento: después de inserción, modificación o borrado en la tabla “Baja_logica”.</p> <p>-Granularidad: una vez por sentencia.</p> <p>-Disparador que actualiza la tabla DW_mejores_equipos con el top 3 de los mejores equipos ofensivos (mayor nº de puntos marcados=) y el top 3 de los mejores equipos defensivos (menor nº de puntos encajados) para cada combinación de competición y temporada.</p> <p>-En caso de empate, el sistema mostrará todos los equipos con el mismo valor de puntos que encajen dentro del correspondiente top 3.</p> <p>-Al tener un carácter histórico, caso de temporadas anteriores, no se tiene en cuenta si el jugador se halla activo o inactivo.</p> <p>-Los top 3 de equipos ofensivos muestran la totalidad de puntos anotados en negativo, mientras que los top 3 de equipos defensivos lo hacen en negativo diferencia entre puntos totales anotados en los partidos en los que el equipo ha participado y los puntos anotados por el equipo en cuestión).</p> <p>-Se comprueban los elementos relacionados con el indicador estadístico (competición, estadística, equipo, partido y contrato) para no incorporar ninguno en baja lógica.</p>
DW_mejor_ju gador	tr_dw_mejor _jug_est	<p>-Evento: después de inserción, modificación o borrado en la tabla “Estadísticas”.</p> <p>-Granularidad: una vez por sentencia.</p> <p>-Disparador que actualiza la tabla DW_mejor_jugador con el jugador con mejor valoración ponderada media para cada combinación de género/modalidad del año (natural no temporada) anterior.</p> <p>-En caso de empate, el sistema mostrará todos los jugadores con el mismo valor de valoración ponderada media más alta.</p> <p>-Al tener un carácter histórico, caso de temporadas anteriores, no se tiene en cuenta si el jugador se halla activo o inactivo.</p> <p>-Se comprueban los elementos relacionados con el indicador estadístico (estadística, jugador, persona, partido y contrato) para no incorporar ninguno en baja lógica.</p>
	tr_dw_mejor_ jug_bajas	<p>-Evento: después de inserción, modificación o borrado en la tabla “Baja_logica”.</p> <p>-Granularidad: una vez por sentencia.</p> <p>-Disparador que actualiza la tabla DW_mejor_jugador con el jugador con mejor valoración ponderada media para cada combinación de género/modalidad del año (natural no temporada) anterior.</p> <p>-En caso de empate, el sistema mostrará todos los jugadores con el mismo valor de valoración ponderada media más alta.</p> <p>-Al tener un carácter histórico, caso de temporadas anteriores, no se tiene en cuenta si el jugador se halla activo o inactivo.</p> <p>-Se comprueban los elementos relacionados con el indicador estadístico (estadística, jugador, persona, partido y contrato) para no incorporar ninguno en baja lógica.</p>

Tabla 20: disparadores módulo estadístico

4.11.2. VALIDACIÓN

Para la verificación del funcionamiento del módulo estadístico se habilitó un juego de pruebas, “pruebas_DW” incluida en la carpeta “SCRIPTS”. Las pruebas incluyeron la ejecución de una consulta de las tablas tras la inserción de datos iniciales en el sistema, comprobando la existencia de información. Los resultados de las consultas correctas fueron contrastados mediante la verificación, por muestreo, de algunos valores de salida que fueron recalculados, o reconsultados, manualmente.

De igual manera, se realizaron bajas lógicas de elementos aparecidos en las tablas de indicadores estadísticos consultadas, y se comprobó el cambio mediante una nueva consulta. Se contrastaron los resultados, comprobando que el elemento dado de baja había modificado el correspondiente indicador estadístico.

Los resultados de la ejecución del juego de pruebas se hallan en el archivo “resultados_pruebas_DW”, incluido en la carpeta “RESULTADOS_PRUEBAS”.

5. VALORACIÓN ECONÓMICA

Los costes del proyecto se basan exclusivamente en las horas de dedicación de los recursos humanos utilizados, ya que no es necesario realizar inversión alguna en hardware, ya disponible, o software, de libre distribución o ya disponible en el caso de MS Office.

Se establecen los costes de 50€/h para el rol de jefe de proyecto, 40€/hora para el de analista, 30€/hora para el de programador y 20€/h para el especialista. El coste total del proyecto es de 5370€

5.1. ESTIMACIÓN INICIAL

La previsión inicial realizada contemplaba las necesidades mostradas en la tabla adjunta, a razón de jornadas de 1,5h diarias.

Nombre de tarea	Días hábiles	Total horas	Rol involucrado
Plan de trabajo	18 días	27	Analista
Análisis de requerimientos	8 días	12	Analista
Diseño	13 días	19,5	Analista
Implementación y pruebas	58 días	87	Programador
Entrega	7 días	10,5	Analista

Tabla 21: detalle de los recursos humanos necesarios iniciales

Lo cual se traduc a en los costes mostrados en la siguiente tabla.

Recurso	Coste (€/h)	Total horas	Coste total (€)
Analista	40	69	2760
Programador	30	87	2610
TOTAL:		5370	

Tabla 22: coste de los recursos humanos iniciales

5.2. COSTE REAL

Tras el desarrollo del proyecto se han detectado necesidades de roles no contemplados en la estimaci3n inicial, as  como un incremento considerable en las horas dedicadas, especialmente durante la fase de implementaci3n.

Nombre de tarea	D�as h�biles	Total horas	Rol involucrado
Plan de trabajo	18 d�as	18+18	Analista / Jefe de proyecto
An�lisis de requerimientos	8 d�as	12	Analista
Dise�o	14 d�as	28	Analista
Instalaci3n , configuraci3n y prueba del entorno de desarrollo	1 d�a	3	Especialista
Implementaci3n y pruebas	62 d�as	186	Programador
Validaci3n al final de cada fase	3 d�as	6	Jefe de proyecto
Entrega	4 d�as	16	Analista

Tabla 23: detalle de los recursos humanos necesarios reales

Los costes reales del proyecto quedan recogidos en la tabla adjunta.

Recurso	Coste (€/h)	Total horas	Coste total (€)
Jefe de proyecto	50	24	1200
Analista	40	56	2240
Programador	30	186	5580
Especialista	20	3	60
TOTAL:		9080	

Tabla 24: coste de los recursos humanos reales

6. CONCLUSIONES

En este apartado se tratan las dificultades halladas, las posibles mejoras detectadas y no implementadas por falta de tiempo, y las conclusiones derivadas de la comparación entre las previsiones iniciales y la realidad

6.1. DIFICULTADES

El control de las convocatorias de jugadores para cada partido, asegurando la presencia de 12 jugadores por equipo y partido ha requerido un control específico. Para ello se ha aprovechado la interrelación “Estadísticas” como gestor de convocatorias, garantizando que el tiempo de juego por equipo no supere el máximo real (5 jugadores en cancha x 40min. de partido=200min.). El estatus de la convocatoria queda reflejado en un atributo informativo de la entidad “Partido” (“est_convocatoria”).

En el mismo sentido el equipo arbitral también ha necesitado un control complejo para asegurar que cada partido disponga de los árbitros requeridos, en función de la competición a la que pertenezca el partido. También actualiza un atributo de estatus en la entidad “Partido” (“est_eq_arb”).

6.2. MEJORAS

6.2.1. CONTROL DE CONVOCATORIAS, ESTADÍSTICAS Y EQUIPOS ARBITRALES

A fin de completar el control de las convocatorias, y por ende de las estadísticas, y de los equipos arbitrales, sería necesario implementar procedimientos de modificación y baja adecuados, ya que únicamente se han implementado los correspondientes al alta. De esta manera se podrá garantizar la correcta actualización de los atributos de estatus de la entidad “Partido”.

Una vez implementados los procedimientos de modificación y baja para las estadísticas/convocatorias y equipos arbitrales, se podrían contemplar los atributos de estatus de “Partido” a fin de descartar partidos con estadísticas/convocatorias incompletas. En la condición actual del proyecto, los atributos de estatus no son tenidos en cuenta para la consulta de partidos y estadísticas. Se deberá tener en cuenta este extremo a la hora de gestionar las estadísticas y los equipos arbitrales de los partidos.

6.2.2. GRANULARIDAD DE LOS DISPARADORES ESTADÍSTICOS

Debido a la restricción de consulta en una tabla afectada por una operación DML (inserción, modificación o borrado), no ha sido posible el uso de disparadores con mayor granularidad (para cada línea en lugar de por sentencia). Este hecho provoca que el sistema deba recalcular todos los indicadores estadísticos relacionados con el disparador ante cualquier evento ejecutado sobre la tabla origen, en lugar de actuar condicionalmente (sólo si el evento afecta realmente al indicador y actualizando sólo la parte afectada).

Esta situación es claramente ineficiente, especialmente para las modificaciones realizadas en la tabla de bajas lógicas, aunque es de esperar que el uso de ésta no sea habitual. Para las tablas origen, tablas del modelo de negocio, es igualmente ineficiente, excepto para aquellos eventos que puedan afectar a diversos registros (por ejemplo modificación de todos los contratos de un equipo concreto).

La solución a esta situación (error de tablas mutantes) pasa por la creación de un doble disparador. El primero, a nivel de fila, almacena los datos a consultar, mientras que el segundo, a nivel de sentencia, realiza la consulta sobre los datos almacenados. Debido al tiempo disponible para la entrega del proyecto se ha desestimado la implementación de esta situación. Así, en caso de que el funcionamiento del sistema se viera realmente afectado por la gruesa granularidad de los disparadores creados, se podría implementar la citada mejora.

6.2.3. GESTIÓN DE BAJAS LÓGICAS

Aunque la implementación de una tabla de registro de bajas lógicas facilita la consulta de elementos en baja lógica, ya que la búsqueda es más inmediata que en las tablas origen de las entidades por la diferencia de volumen de datos a gestionar, y ahorra el espacio de disponer de un atributo por entidad, también comporta que en algunos casos las consultas se dupliquen, ya que se hace necesario realizar la búsqueda en la tabla origen y también en la tabla de bajas.

En este caso habrá que valorar el impacto de este hecho sobre el rendimiento del sistema, teniendo en cuenta que la mayor parte de las consultas implementadas requieren de esta información para su toma de decisiones.

La solución pasaría por incorporar un atributo indicador de estatus para cada entidad del modelo de negocio.

6.3. CONCLUSIONES

Todo y que se ha intentado realizar un diseño exhaustivo y detallado, siendo consciente de la importancia de la fase de diseño, es fácil dejar algún elemento sin considerar. Por tanto se hace más patente, si cabe, la necesidad de realizar un diseño extremadamente detallado para evitar complicar la fase de implementación.

De igual manera, la fase de implementación presenta imponderables que dilatan su duración considerablemente, especialmente si no se dispone de experiencia en los elementos a implementar.

Los costes del proyecto han aumentado considerablemente respecto a la previsión inicial (ver punto 5 de esta memoria) debido al aumento de horas dedicadas. Será necesario, en lo sucesivo, cotizar un tiempo de margen ante imprevistos, especialmente si no se dispone de experiencia en alguno de los elementos a incluir en el proyecto.

El presente TFC ha constituido un excelente banco de pruebas para adquirir experiencia y nuevas competencias en la gestión de proyectos de creación de software. También ha facilitado la adquisición de nuevos conocimientos referidos al lenguaje PL/SQL, y en general una toma de contacto con el sistema gestor Oracle.

7. GLOSARIO

Entidad: representación de un concepto de la realidad en la base de datos. Habitualmente conocido como “tabla”.

Atributo: características que definen a las entidades. Habitualmente conocido como “campo”.

Base de datos relacional: base de datos en la cual los elementos se relacionan a través de atributos vinculados que disponen del mismo valor.

Interrelación: relación entre diversas entidades de una base de datos.

Modelo de negocio: elementos de la base de datos relacionada con las particularidades de la realidad a gestionar.

Procedimiento almacenado: programa almacenado en la base de datos que puede ser ejecutado a partir de una llamada, en la que se incluirán unos determinados parámetros de entrada, y que ejecuta diversas instrucciones de manipulación de datos.

Disparador: programa almacenado en la base de datos que se ejecuta automáticamente cuando se cumplen una o varias condiciones, ejecutando diversas instrucciones de manipulación de datos.

El resto de conceptos aparecidos ya fueron descritos a lo largo del presente documento, a medida que fueron apareciendo.

Conectividad: define el tipo de correspondencia entre las ocurrencias de entidades asociadas mediante una interrelación.

8. BIBLIOGRAFIA

8.1. LIBROS

- Módulo “*Recollida i documentació de requisits*” de la asignatura “*Enginyeria del programari*” de la UOC.
- Módulo “*Administració de les dades*” de la asignatura “*Administració de sistemes operatius i xarxes*” de la UOC.
- Módulo “*Disseny de bases de dades*” de la asignatura BDI de la UOC.
- Módulo “*Teoria de la normalització*” de la asignatura BDI de la UOC.
- Módulo “*Introducció a les bases de dades*” de la asignatura BDI de la UOC.
- Teaching Soft Group, “Oracle 11g, Curso práctico”, Ra-Ma. Madrid. 2011.
- Teaching Soft Group, “Programación en Oracle 11g, SQL, SQL*PLUS y PL/SQL”, Ra-Ma. Madrid. 2011.
- Francisco Charte Ojeda, “Guía Práctica SQL”, Anaya Multimedia, Madrid. 2010.

8.2. WEBS MODELO DE NEGOCIO

- Países sin código postal:
http://es.wikipedia.org/wiki/C%C3%B3digo_postal
- Federación Española de Baloncesto:
www.feb.es
- Municipios con nombres repetidos en España:
<http://www.albaiges.com/eulogologia/ecos/pueblosnombrerepetidoespana.htm>

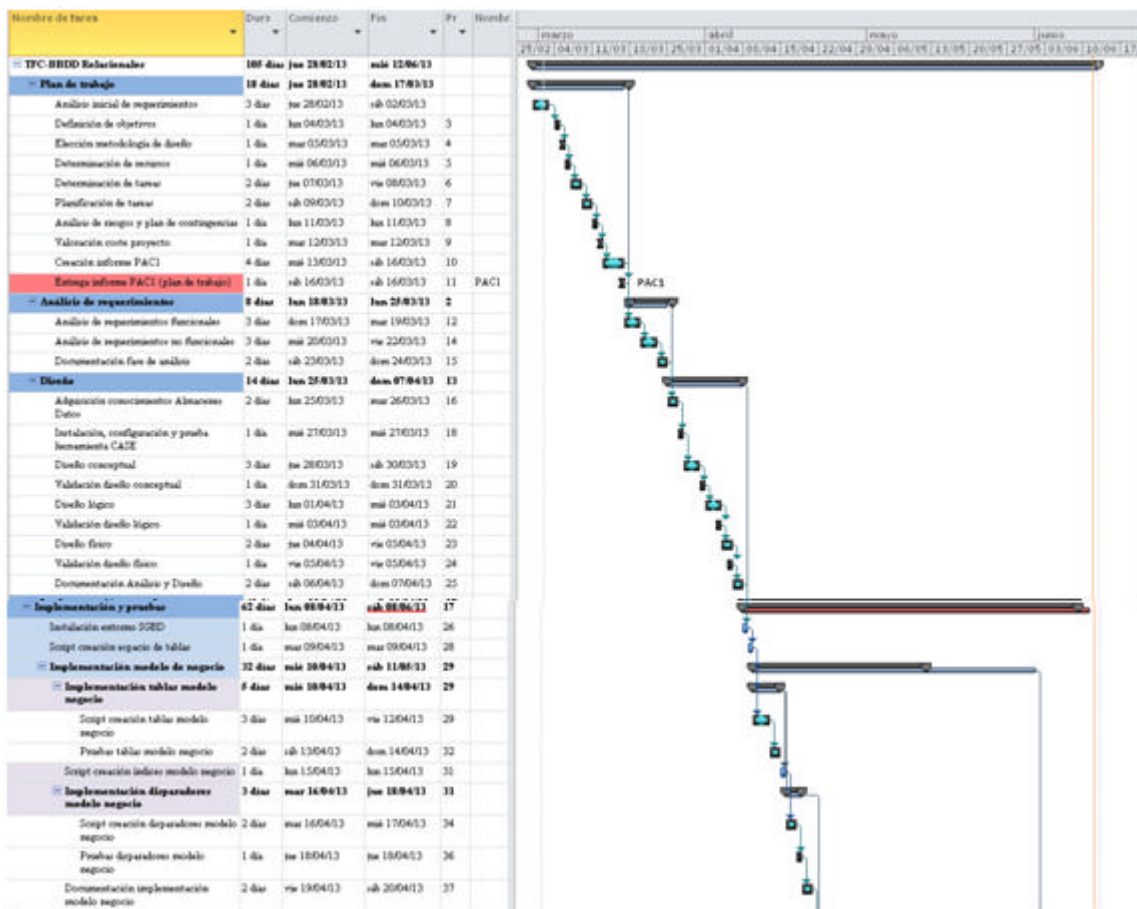
- Municipios con nombres repetidos en todo el mundo:
<http://www.skyscrapercity.com/showthread.php?t=414176>
- Duración ronda clasificatoria para el mundial de baloncesto:
http://es.wikipedia.org/wiki/Campeonato_Mundial_de_Baloncesto_de_2014

8.3. WEBS ÁMBITO TÉCNICO

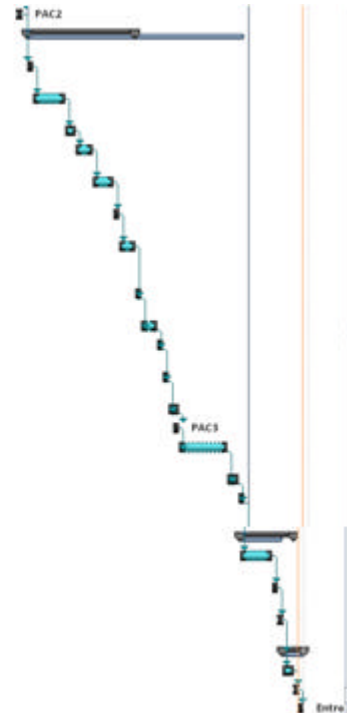
- Creación de índices :
<http://www.oracleya.com.ar/temarios/descripcion.php?cod=203&punto=45>
- Error tabla mutante en disparadores:
<http://dis.um.es/~jfernand/0405/dbd/DBD-TRIGGERS.pdf>
- Documentación Oracle 11g R2 Express Edition:
http://docs.oracle.com/cd/E17781_01/install.112/e18803.pdf
- Documentación SQL Developer 3.2.20.09.87:
http://docs.oracle.com/cd/E35137_01/doc.32/e35119.pdf
- Conversión implícita de tipos realizada por Oracle:
http://www.mundoracle.com/funciones-sql.html?Pg=sql_plsql_3.htm
- Funciones analíticas:
http://www.epidataconsulting.com/tikiwiki/tiki-read_article.php?articleId=64

9. ANEXOS

Planificación definitiva:



Entrega informe PAC2	1 día	sáb 20/04/13	sáb 20/04/13	38	PAC2
Implementación procedimientos módulo negocio	20 días	lun 22/04/13	sáb 11/05/13	38	
Agrupación conocimientos tratamiento excepciones	1 día	lun 22/04/13	lun 22/04/13	39	
Script creación procedimientos ABM	6 días	mar 23/04/13	dom 28/04/13	41	
Pruebas procedimientos ABM	2 días	lun 29/04/13	mar 30/04/13	42	
Introducción datos en tabla modelo negocio	3 días	mié 01/05/13	vie 03/05/13	43	
Script creación procedimientos estadísticas partidos	4 días	sáb 04/05/13	mar 07/05/13	44	
Pruebas procedimiento estadísticas partidos	1 día	mié 08/05/13	mié 08/05/13	45	
Script creación procedimiento equipo arbitral	3 días	lun 09/05/13	sáb 11/05/13	46	
Pruebas procedimientos equipo arbitral	1 día	dom 12/05/13	dom 12/05/13	47	
Mejoras del sistema	3 días	lun 13/05/13	mié 15/05/13	48	
Creación disparadores control bonificación	1 día	lun 14/05/13	lun 14/05/13	49	
Pruebas disparadores control bonificación	1 día	vie 17/05/13	vie 17/05/13	50	
Creación informe PAC3	2 días	sáb 18/05/13	dom 19/05/13	51	
Entrega informe PAC3	1 día	dom 19/05/13	dom 19/05/13	52	PAC3
Script creación procedimientos de consulta	9 días	lun 20/05/13	mar 28/05/13	53	
Pruebas procedimientos de consulta	2 días	mié 28/05/13	lun 30/05/13	54	
Documentación procedimientos de consulta	1 día	vie 31/05/13	vie 31/05/13	55	
Implementación Módulo Estadístico	18 días	sáb 01/06/13	lun 18/06/13	56	
Script creación disparadores módulo estadístico	6 días	sáb 01/06/13	lun 04/06/13	56	
Pruebas disparadores módulo estadístico	1 día	vie 07/06/13	vie 07/06/13	58	
Documentación implementación módulo estadístico	1 día	sáb 08/06/13	sáb 08/06/13	59	
Entrega	4 días	dom 09/06/13	mié 12/06/13	60	
Revisión memoria proyecto	2 días	dom 09/06/13	lun 10/06/13	60	
Creación presentación proyecto	1 día	mar 11/06/13	mar 11/06/13	62	
Entrega final del proyecto	1 día	mié 12/06/13	mié 12/06/13	63	Entrega



Scripts de implementación y pruebas (carpeta Producto):

- “creación_tablespaces”.
- “creación_indices”.
- “creación_tablas”.
- “creación_secuencias”.
- ”creación_usuarios”.
- “creación_disparadores”.
- “ABM_alta_partido”
- “ABM_modif_partido”
- “ABM_baja_partido”
- “ABM_alta_contrato”
- “ABM_modif_contrato”
- “ABM_baja_contrato”
- “ABM_alta_jugador”
- “ABM_modif_jugador”
- “ABM_baja_jugador”
- “alta_eq_arbitral”
- “alta_estadísticas”
- “consulta_A”
- “consulta_B”
- “consulta_C”

- “consulta_D”
- “consulta_E”
- “consulta_F”
- “DW_1”
- “DW_2”
- “DW_3”
- “DW_4”
- “DW_5”
- “DW_6”
- “DW_7”
- “implementación_sistema_SYS
- “implementación_sistema_administrador”
- “introducción_datos_iniciales”
- “pruebas_consultas”
- “pruebas_DW”
- “pruebas_procs_y_disp_her”

Archivos de resultados de pruebas (carpeta resultados pruebas):

- “resultados_pruebas_consultas”.
- “resultados_pruebas_disp_her_disj”.
- “resultados_pruebas_DW”
- “resultados_pruebas_procs”

Instrucciones de instalación (archivo .txt en carpeta Producto).