

Sistema de Información

“SmartyPadel”

“Tu buscador de palas de pádel personalizado al mejor precio”.
Ingeniería de Sistemas de Información.



Daniel Bolaños Martínez
Miguel Ángel Cantarero López

Índice

1. Propuesta del proyecto de prácticas.....	3-4
Título provisional del proyecto.....	3
Integrantes del equipo de desarrollo.....	3
Descripción del ámbito del proyecto.....	3
Proyectos y sistemas existentes en el mismo ámbito.....	4
Lista de fuentes de datos que se pretenden utilizar.....	4
2. Presupuesto del sistema.....	5-9
Presupuesto de desarrollo del sistema.....	5-7
Flujo de caja asociado al sistema.....	7
Presupuesto de despliegue del sistema.....	8-9
Versión “in house”.....	8
Versión basada en la nube.....	9
Referencias.....	9
3. Diseño arquitectónico.....	10-13
Descripción de la arquitectura.....	10
Diagramas de diseño del sistema.....	11-13
Referencias.....	13
4. Versión Inicial del Sistema.....	13

1. Propuesta del proyecto de prácticas.

Título provisional del proyecto.

SmartyPadel. (Buscador de palas de pádel personalizado al mejor precio).

Integrantes del equipo de desarrollo.

El equipo está integrado por:

Daniel Bolaños Martínez.

Miguel Ángel Cantarero López.

Descripción del ámbito del proyecto.

El objetivo principal de nuestra aplicación es ayudar al jugador de pádel, experto o no, en la compra de una pala de padel al menor precio posible; así como poder comparar distintas opciones de palas según las características de las mismas.

Para lograrlo, implementaremos un comparador de precios especializado que muestra la mejor oferta de palas para un jugador según sus características de juego.

De esta manera, el cliente no tiene que adaptarse al producto que va a adquirir, sino que nuestro software le ofrecerá una gama de palas que mejor se adapten a lo que esté buscando. Una vez el cliente se decanta por un modelo, nuestro buscador ofrecerá la opción de menor precio en la red, comparando entre las diferentes fuentes de datos.

En el mercado encontramos varios sistemas encargados de la venta de material de pádel, pero nuestro sistema marca la diferencia a la hora de ofrecer la opción que mejor se adapta al cliente, mientras que en el resto, es el cliente el que tiene que realizar una búsqueda manual y sin conocimiento sobre el rendimiento del producto que va a adquirir.

Además ofrecemos una amplia descripción del producto que se adapta a la jugabilidad de cada persona (potencia, manejo...), así como una ordenación por relevancia destacando los modelos con mejores reseñas de profesionales.

Proyectos y sistemas existentes en el mismo ámbito.

Aquí podemos observar tres páginas webs que utilizan un servicio similar al que queremos implementar. Estas aplicaciones, ofrecen en su página de ventas un comparador de precios por modelo de los productos que ellos mismos venden.

paddelea.com/palas-padel/comparador/
padelavenue.es/comparar-palas-padel.html
padelhome.es/comparador-palas-de-padel/

Lista de fuentes de datos que se pretenden utilizar.

Algunas de las fuentes de datos de las que extraemos la información para nuestra aplicación, son:

Base de datos de palas de pádel de las que obtendremos el modelo y marca de la pala para posteriormente hacer la búsqueda de la mejor oferta.

padelzoom.es

Webs y APIs utilizadas para buscar el producto que se desea comprar y de las que también obtendremos los datos y puntuaciones de los mismos.

amazon.com
time2padel.com
padelsuite.com

2. Presupuesto del sistema.

Simulación aproximada del presupuesto para el desarrollo del sistema de información propuesto: “SmartyPadel”. Teniendo en cuenta el esfuerzo estimado para la realización del proyecto como el coste asociado a la puesta en marcha del mismo. Así como los costes del despliegue del sistema en dos versiones: “in house” y contratando un servicio de “cloud computing”.

Presupuesto de desarrollo del sistema.

Gastos de personal:

Realizaremos la estimación del salario a partir de un simulador online que calcula a partir del sueldo bruto y aplicando los tipos impositivos vigentes de acuerdo a nuestra situación, el sueldo neto. [1]

Plantilla: 2 trabajadores.

- Situación familiar: Soltero. Sin hijos.
- Edad: 22.
- Categoría profesional: Ingenieros y Licenciados.

Situación Laboral por persona (suponiendo duración de 1 año de proyecto):

- Tipo de contrato: Indefinido.
- Número de pagas: 12.
- Sueldo bruto anual: 25.000,0€
- Retenciones por IRPF: 3.532,5€
- Cuotas a la Seg. Social: 1.587,5€
- Sueldo neto anual: 19.880,0€
- Tipo de retención sobre la nómina: 14,13%
- Sueldo neto mensual (12 pagas): **1.656,7€**

Porcentaje de cotización de la empresa por trabajador: 30,9% del sueldo bruto por contratos indefinidos. (Contiene cotizaciones por contingencias comunes, desempleo, accidentes y enfermedad, formación profesional y FOGASA). [2]

Por tanto supone un coste a la empresa anual de:

50.000€ (sueldo bruto 2 trabajadores) + 15.450€ (porcentaje añadido) = **65.450€**

Suponemos una duración del proyecto de 4 meses, obtenemos que el coste personal de la empresa para este proyecto es de: **21.816,67€**

Gastos de ejecución:

- **Costes de adquisición de material inventariable.**

- (x2) Acer Aspire E15 (Intel Core i7-6500U, 8 GB RAM, 1 TB HDD, Nvidia GTX 950M de 2 GB, Windows 10): **1.600 €**
- (x1) HP LaserJet Pro MFP M281fdw Multifunción Láser Color: **297,3€**
- (x1) Teléfono Fijo Grandstream GXP1625: **35,95€**
- (x2) Teléfono móvil Samsung Galaxy J7: **399,80€**
- (x1) Seagate HDD 4 TB (2,5', 15 mm, 128 MB de caché SATA 6 GB/s hasta 140 MB/s): **149,99€**

El material informático es amortizable al 25% anual (con una vida útil de 4 años).

Gastos en material inventariable (100%a amortizado) = **2483.04€**

Porcentaje relativo a 1 año = **620.76€**

Porcentaje relativo a los 4 meses de duración del proyecto = **206,92€**

- **Costes de adquisición de material fungible.**

- Material de oficina (folios, bolígrafos, grapas...): **53,67€**
- (x4) Pack de 2 cartuchos de tinta original HP negro/Tri-color: **139,96€**

Total = **190,63€**

- **Costes de contratos, patentes, licencias, consultoría, suministros...**

- (x2) Tarifa móvil 25 GB y llamadas ilimitadas Vodafone (Alta gratis): **50€/mes**
- (x1) Fibra Óptica 50 MB simétricos + Fijo ilimitado: **18,15€** (alta) + **12,85€/mes**
- (x1) Licencia Atlassian para desarrollo software hasta 10 usuarios: **8,82€/mes**
- (x1) Licencia de uso de base de datos NoSQL: **0€**

Total (en los 4 meses de desarrollo del proyecto) = **304,83€**

Gastos complementarios:

- **Gastos de desplazamiento, viajes, estancias y dietas.**

- (x1) Estancia en Barcelona durante del curso (Hotel 3 estrellas, Sitges):
1.381€

Total = **1.381€**

- **Gastos de material de difusión, publicaciones, promoción, catálogos, folletos, cartelería... [3] [6]**

- (x1) Diseño de logotipo para el proyecto: **150€**
- (x1) Diseño cartel A3 una cara para publicitar en torneos de pádel: **85€**
- (x50) Impresión de los carteles en formato físico: **32,52€**
- (x1) Google AdWords: **200€**
- (x1) Facebook Ads: **75€**

Total = **542,52€**

- **Gastos de inscripción en cursos, congresos y seminarios relacionados con el proyecto. [4]**

- (x1) Curso de formación realizado por técnico docente: **36€/hora**
- (x1) Curso presencial Big Data For Sales en Barcelona: **650€**

Total = $36€ \cdot (25 \text{ horas presenciales}) + 650€ =$ **1.550€**

Flujo de caja asociado al proyecto:

El total de gastos del desarrollo del proyecto en los 4 meses de duración sumando los gastos de personal y los de ejecución, asciende a un total de **25.992,57€**

Presupuesto de despliegue del sistema.

Para el cálculo del despliegue del sistema supondremos para ambas versiones una duración aproximada de 1 año de funcionamiento de la aplicación.

Tendremos en cuenta también el precio del material fungible e inventariable para obtener toda la infraestructura necesaria para el despliegue del proyecto.

El coste del material inventariable (compra de material informático) para 1 año de duración del proyecto asciende a **620.76€**, y el coste de los contratos y licencias para poder desarrollar el software en el mismo tiempo, asciende a **154,2€** (tenemos solo en cuenta la tarifa de internet, sin el precio de alta que se supone pagado al iniciar el proyecto).

- **Sistema “in house”:**

- Servidor torre PowerEdge T630: **1.550,22 €**
- Dominio web [7]: **1.99€**
- Mantenimiento y monitorización del servidor: **229€/mes**
- Software Apache HTTP server: 0€
- Sistema operativo linux 18.04 LTS: 0€
- Coste de luz: Para el cálculo de coste de luz , consideramos como gasto base el del servidor que permanece 24h al día en funcionamiento, tiene una potencia de 750 W, teniendo en cuenta los precios medios de tarifa durante el día 0,104€/kWh y la tarifa nocturna 0,123€/kWh. El router puede tener una potencia de 25W y los portátiles 45W. Con unos cálculos simples, y suponiendo que contratamos una potencia de 31,2 kW. La tarifa de la luz asciende como mínimo a **180€/mes**

Coste de ejecución (1 año): **774,96€**

Coste despliegue del sistema (1 año): **6.460,21€**

Coste total del despliegue (1 año): 7.235,17€

Coste primer mes de despliegue: 602,93€

- **Sistema basado en la nube:**

Para obtener un menor tiempo de respuesta en cloud computing, es aconsejable comprar los servicios de un servidor en la nube que nos proporcione menor latencia con respecto a la zona geográfica en la que trabajemos. En este caso, hemos

hechos los cálculos con el servidor de Londres (europe-west2) que ofrece Google Cloud que es más caro que el de Bélgica, pero ofrece menor tiempo de respuesta.
[5]

- Hosting Google Cloud SQL: Almacenamiento HDD 60GB: **5,76€/mes**
- Hosting Google Cloud SQL: Backups (Used) 120GB: **10,2€/mes**
- Memoria 8GB alta disponibilidad: **87,28€**
- **(x4)** vCPUs de alta disponibilidad: **257,52€**
- Sistema operativo linux 18.04 LTS: **0€**
- Mantenimiento y monitorización del sistema: **149€/mes**

Coste de ejecución (1 año): **774,96€**

Coste despliegue del sistema (1 año): **2.324,32€**

Coste total del despliegue (1 año): 3.099,28€

Coste primer mes de despliegue: 258,27€

Podemos concluir con que el precio de despliegue del sistema “in house” es casi el triple que el del sistema empleando “cloud computing”, esto es bastante lógico, pero probablemente el servidor “in house” será mucho más eficiente y producirá menor latencia que el basado en la nube, aunque hayamos invertido un dinero extra en adquirir servicios de alta disponibilidad que mejoran el rendimiento del servidor en la nube. Dependiendo del capital inicial de nuestra empresa, nos decantaríamos por una opción u otra.

Referencias.

[1] https://cincodias.elpais.com/herramientas/calculadora-sueldo-neto/#tabla_resultados

[2] <https://infoautonomos.eleconomista.es/blog/cuanto-cuesta-contratar-un-trabajador/>

[3] <http://www.inventanova.com/tarifas-diseno-grafico/>

[4] <https://www.emagister.com/cursos-big-data-barcelona-kwprov-10000053189-9.htm>

[5] <https://cloud.google.com/sql/pricing>

[6] <https://ads.google.com/home/>

[7] <https://www.strato.es/buy/spa/specialoffer/1>

3. Diseño arquitectónico.

Descripción de la arquitectura:

Decisión de diseño de la arquitectura: Arquitectura 3 capas.

Justificación de la elección: Utilizaremos una arquitectura multinivel ya que nos permite distribuir los roles y responsabilidades del sistema de forma jerárquica separando las responsabilidades. El rol indica el modo y tipo de interacción con otras capas y la responsabilidad indica la funcionalidad que está siendo desarrollada.

Las tres capas implementadas serán: una capa de presentación (funcionalidad relacionada con la interfaz de usuario), una capa de negocio (funcionalidad relacionada con la aplicación) y una capa de datos (funcionalidad relacionada con el acceso a datos).

En nuestro caso implementaremos la capa de presentación en forma de aplicación web con interfaz de usuario, la capa de negocio con soporte en Google Engine y la capa de datos como una base de datos NoSQL en una memoria caché.

Implementando la capa de negocio con cloud computing nos da la ventaja de implementar el servicio sin necesidad de que el cliente requiera inversión previa y la base de datos NoSQL nos ofrece escalabilidad horizontal, manejo de gran cantidad de recursos y no genera cuellos de botella frente al modelo relacional SQL. De esta forma podremos implementar un modelo de integración para las diversas fuentes de datos.[1]

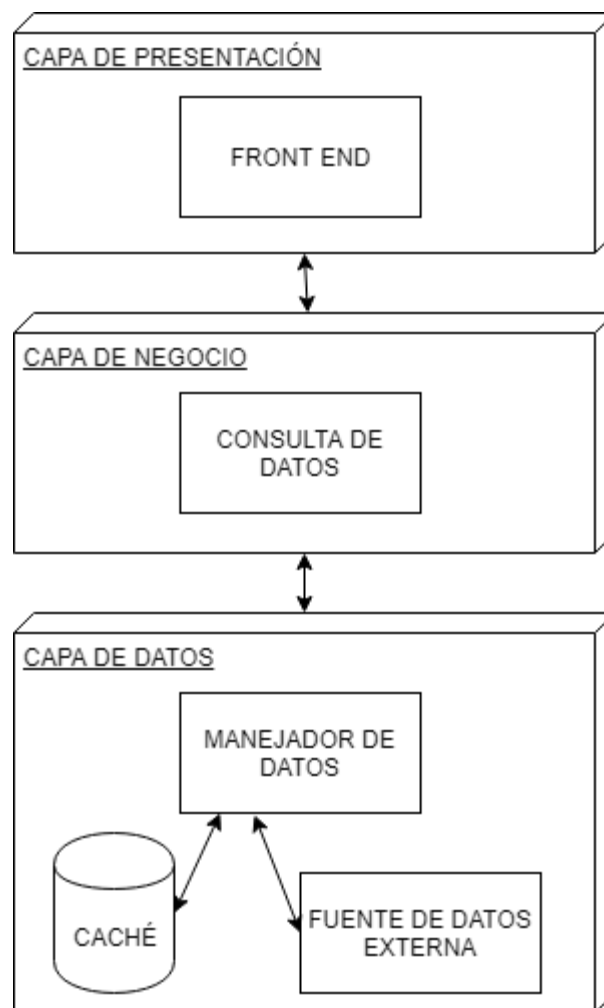
Alternativas de diseño consideradas:

- Modelo cliente-servidor: descartado por sus limitaciones de escalabilidad y extensibilidad. El servidor supone un punto único de fallo.
- Modelo centralizado: descartado por el alto coste de adquisición y renovación. Crea gran dependencia entre los usuarios con el sistema. Sistema rígido difícilmente actualizable.
- Modelo relacional SQL: descartado por sus limitaciones en escalabilidad (cuello de botella). Deficiencias en el procesamiento de datos a tiempo real.[2]

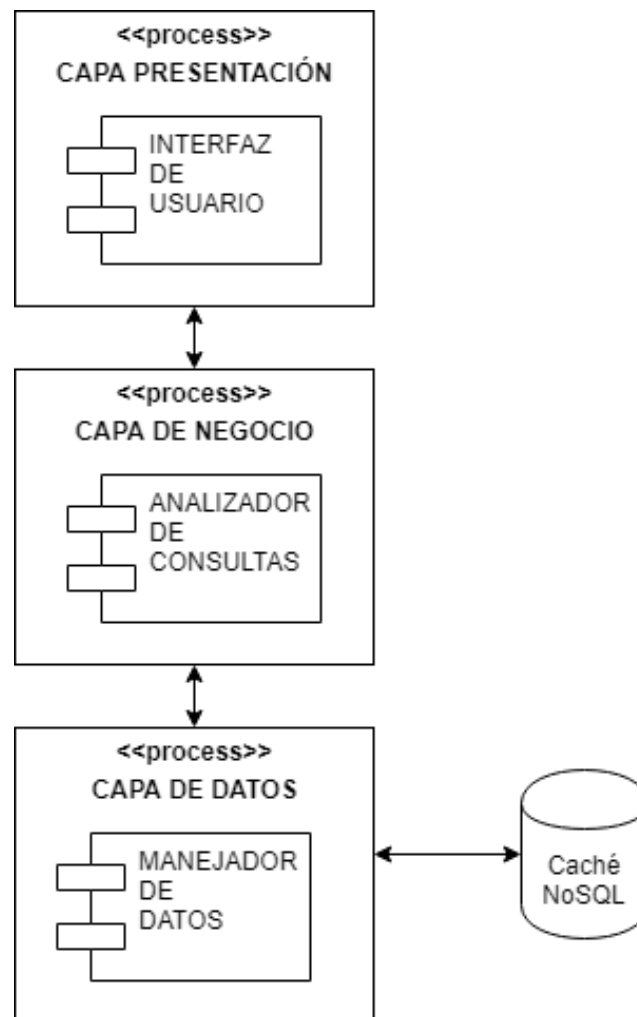
Diagramas de diseño del sistema:

Usaremos el Modelo 4+1 de Vistas para ilustrar el diseño de nuestro sistema. Obviamos la vista física (no sabemos como trabaja el hardware de Google Engine), así como el modelo de casos de uso.

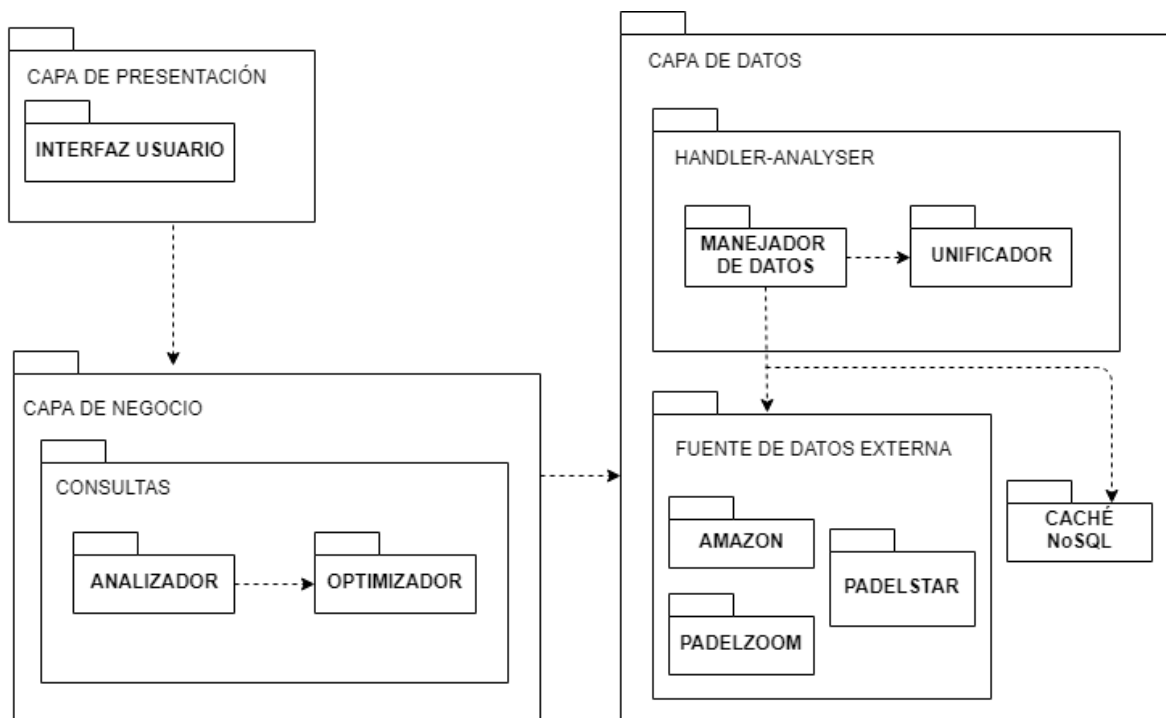
Vista lógica: (Modelo OO del diseño). Haremos la división en tres capas. La capa de presentación que será el front end del sistema, la de datos que tratará los datos del sistema y la de negocio que se encargará de manejar la comunicación entre las dos anteriores.



Vista de procesos: (Concurrencia y sincronización). Explicación de los procesos del sistema y la comunicación entre ellos. Tendremos un proceso por capa: Para la capa de presentación uno encargado de atender a los clientes. Para la de negocio uno que analice y optimice las consultas y para la de datos uno que acceda a los datos sobre la caché o a una fuente de datos externa.



Vista de desarrollo: (Organización de los módulos). La capa de presentación contendrá la interfaz de usuario del sistema. La capa de negocio estará orientada al módulo de consultas en la que habrá un analizador de las consultas que asegure la consistencia de los datos y un optimizador para que el tiempo de búsqueda sea eficiente. En la capa de datos tendremos un módulo con las fuentes de datos externas, otro con la caché y finalmente uno que actúe como manejador y unifique los datos de las distintas fuentes.



Referencias.

[1] <https://geeks.ms/jkpelaiez/2009/05/30/arquitectura-basada-en-capas/>

[2] <https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf>

4. Versión Inicial del sistema.

Enlace de la URL pública que contiene nuestro Sistema de Información.

<https://smartypadel.appspot.com>