



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

TESIS DEL MÁSTER EN INGENIERÍA DEL SOFTWARE, MÉTODOS FORMALES Y  
SISTEMAS DE INFORMACIÓN

# PLATAFORMA WEB DE CONTRATACIÓN DE PRODUCTOS HOTELEROS

Rafael Esteller Blasco

Dirigida por Dr. Juan Sánchez Díaz  
Tutorizada por Vicent González i Castells

# Índice de contenido

1. Introducción.....	4
2. Motivación.....	5
3. Modelado de requisitos.....	6
3.1 Descripción del proceso de negocio de alto nivel.....	7
3.2. El modelo de datos.....	9
4. Desarrollo de la aplicación.....	10
4.1 Metodología de desarrollo.....	10
4.2 Entorno de desarrollo.....	17
4.2.1 Tecnología.....	17
4.3 Arquitectura de la aplicación.....	18
5. Demostración de la aplicación.....	20
5.1 Buscador.....	20
5.3 Listado de artículos disponibles.....	21
5.4 Página de reserva ( I ).....	22
5.5 Página de reserva ( II ) .....	23
6. Conclusiones.....	24
Apéndices.....	26
Apéndice A. Modelo de procesos de negocio (I).....	26
Apéndice B. Modelo de datos del proveedor.....	28
Bibliografía.....	29

*El desarrollo del proyecto que este documento presenta se ha llevado a cabo en la empresa **programadorphp.es**, y se enmarca dentro un periodo de estancia en prácticas realizadas a través del Servicio Integrado de Empleo de la UPV. Esta empresa está especializada en desarrollo web, aplicaciones empresariales, e-commerce y desarrollo de aplicaciones para dispositivos móviles.*

*A petición del cliente, no se expondrá código fuente de la aplicación en este documento. Sin embargo, se podrán contrastar algunas de las funcionalidades a las que se hace referencia accediendo a [www.roomsday.com](http://www.roomsday.com)*

# 1. Introducción

El presente documento pretende explicar y analizar las distintas fases del desarrollo de un proyecto web llevado a cabo en un entorno industrial.

En el primer capítulo se describirá la principal motivación y objetivos a cubrir por el sistema, siendo el objetivo general el construir una plataforma de reserva directa de productos hoteleros para una agencia de viajes online. El sistema debe dar soporte al proceso completo de búsqueda, reserva y cancelación de productos hoteleros, los cuales se obtendrán a través de diferentes proveedores mayoristas.

Continuaremos analizando estos objetivos a través de los modelos elegidos para extraer los requisitos del sistema. En este caso la extracción de requisitos se ha llevado a cabo en base al estudio de un modelo de procesos de negocio y a un modelo de datos.

Dada la naturaleza del proyecto cobra gran importancia la definición del proceso de negocio, pues la plataforma actúa de intermediaria entre clientes y proveedores, y de su interacción surgen flujos de información que requieren un análisis detallado. Veremos cómo se ha plasmado este proceso de negocio en un diagrama BPMN al que acompaña de una definición textual.

De igual modo, es especialmente relevante el modelo de datos utilizado para dar soporte al sistema. El diseño de este modelo es fruto del análisis de las tareas y flujos de información generados en el proceso de negocio, así como de las estructuras de datos que proporcionan los proveedores.

Pasaremos entonces a analizar los detalles relativos al propio desarrollo la aplicación, donde se explicará qué motivaciones han llevado a la elección de una metodología ágil como el prototipado. Respaldaán esta decisión factores como el tamaño y distribución del equipo de desarrollo, así como los roles asignados.

Analizaremos de qué forma se ha llevado a cabo el diseño de la aplicación para permitir la integración con diferentes proveedores de productos hoteleros y se estudiará en detalle la integración en un caso concreto. Expondremos los modelos de datos local y del proveedor, así como el proceso de sincronización entre ambos.

## 2. Motivación

Empresa del sector vacacional, que actualmente dispone de diferentes plataformas de contratación de hoteles para grupos a través de agencias de viaje, apuesta por la apertura de una nueva plataforma de contratación hotelera online de venta directa al público: roomsday.com.

Dada la competencia existente en el sector, se busca el factor diferencial en la facilidad y agilidad del entorno de búsqueda y reserva, así como en ofrecer precios muy competitivos en productos seleccionados.

El desarrollo de esta plataforma plantea un diseño complejo, pues ha de proporcionar la posibilidad de integración con múltiples mayoristas. Esta integración debe ser transparente al usuario tanto desde el punto de vista de los contenidos como de los procedimientos a seguir para completar los procesos de negocio (búsqueda, pre-reserva, reserva, cancelación, pago, etc).

La **integración de la información** expuesta por los proveedores plantea retos a nivel de:

- Organización y estructuración de la información: Cada proveedor organiza la información relativa a sus artículos en base a unas estructuras y criterios que no tienen por qué coincidir con el resto de proveedores. Esta información deberá ser tratada con el fin de poder presentarla a los usuarios de una forma homogénea. Se hacen necesarias unas estructuras compatibles con todos los proveedores.
- Sincronización: La aplicación debe actualizar de forma periódica los artículos que ofrece y la información auxiliar que da soporte a los procesos de la misma. Esta actualización debe tener en cuenta y respetar las posibles modificaciones realizadas por el personal de la agencia sobre la información local.

La **integración de los procedimientos** del proceso de negocio de la aplicación con las operaciones expuestas por los proveedores plantea la necesidad de definir un diseño que homogenice las interfaces de los proveedores para así facilitar el desarrollo del resto de capas de la aplicación.

### 3. Modelado de requisitos

Un factor clave para el éxito en la construcción de un sistema de información es la correcta adquisición de los requisitos que ha de satisfacer el mismo. Con este fin, hemos considerado estudiar:

- La organización que lo explotará y las actividades que debe cubrir. *El proceso de negocio.*

*“Un proceso de negocio consiste en un conjunto de actividades que se desarrollan coordinadamente en un entorno organizacional y técnico. Estas actividades satisfacen conjuntamente alguna meta de negocio. Cada proceso de negocio se desarrolla en una sola organización, pero puede interactuar con procesos de negocio de otras organizaciones”<sup>1</sup>.*

Mediante el estudio del proceso de negocio podremos extraer los flujos que tomará la información, quién tendrá acceso a ella y en qué momento ésta debe ser accesible. Esta tarea facilitará la extracción de requisitos a todos los niveles.

Un modelo comprensible por todas las partes implicadas en el desarrollo del sistema facilitará a cada una de ellas trasladar su visión del mismo y facilitar la discusión sobre aquellas partes que no queden completamente definidas en las especificaciones textuales.

Como veremos más adelante, la construcción del modelo de procesos ha sido llevada a cabo mediante diagramas BPMN (Business Process Modeling Notation).

Se ha considerado oportuna la utilización de dicha notación por diferentes motivos:

- Es sencilla de interpretar por personas con perfiles técnicos o perfiles organizacionales (no técnicos).
- Facilidad de acceso a documentación. Gran cantidad de recursos en Internet y en bibliografía especializada.
- Permite definir modelos a diferentes niveles de detalle.
- Existen herramientas de libre distribución que permiten la edición de modelos con esta notación. Para este trabajo, se ha utilizado la herramienta *Yaoqiang v.2.2*.

---

1 Definición por *Weske*, 2005.

### 3.1 Descripción del proceso de negocio de alto nivel

La descripción que se adjunta a continuación tiene como finalidad complementar el diagrama BPMN del Apéndice A.

Definimos a continuación los actores que participan en el proceso.

“Cliente”:

Usuario beneficiario de los servicios que ofrece la plataforma de reservas.

“Proveedor”:

Expone mediante operaciones de servicios web información y funcionalidad referente a sus productos ofertados.

“Agente comercial”:

Atiende las incidencias de clientes a través del teléfono o email.

“Agencia de viajes”

Ejerce de intermediaria entre mayoristas y consumidores. Define qué información y cómo se va mostrar ésta en la web: marcas, rankings ofertas, criterios de ordenación de resultados, etc.

El proceso de negocio da comienzo en la recepción de una búsqueda en el sistema hecha por el cliente. Esta búsqueda inicial requiere de al menos los siguientes parámetros:

- Hotel / Ciudad / Marca/ Zona (Guiados por el campo con función autocompletar)
- Fechas de entrada y salida
- Número de habitaciones
- Número de adultos y número de niños por habitación.

El sistema realiza una primera búsqueda en sus índices de hoteles, ciudades y marcas (una marca agrupa hoteles o ciudades, p.ej. “Costa blanca”) tras la cual se lanzan las consultas de disponibilidad a todos aquellos proveedores que puedan contener productos para los parámetros dados.

Una vez los proveedores hayan devuelto todos sus resultados, o se haya llegado a un tiempo máximo de espera, el sistema mostrará los datos de

disponibilidad al cliente para los parámetros de entrada introducidos. En caso de producirse algún error durante la búsqueda de disponibilidad para este destino, se informará al cliente del error y se le presentarán destinos alternativos.

En el momento en que se retornen los resultados de disponibilidad se iniciará una “sesión” que debe expirar a los 30 minutos, pues la disponibilidad se consulta en tiempo real, y existen restricciones en el tiempo de validez de los resultados obtenidos en esta consulta.

El cliente podrá seleccionar entonces un artículo y acceder a los detalles de su ficha, donde se detalla toda la información referente a las características del producto, precio, régimen, tipo de habitación y comodidades de la misma.

Si el cliente decide reservar dicho producto, el sistema debe enviar una orden de pre-reserva al proveedor, de modo que el producto siga estando disponible durante la introducción de los datos de reserva y hasta el momento de su reserva y pago.

En esta primera pre-reserva se iniciará otra sesión, que expirará en 30 minutos, siendo éste el tiempo máximo de vigencia de una pre-reserva antes de su confirmación.

En caso de no haber disponibilidad en el momento de la llamada de pre-reserva, el proveedor lo notificará y el sistema avisará al usuario para que se realice una nueva búsqueda o sugerirá artículos similares al elegido.

Para hacer efectiva la reserva se le presentará al cliente una serie de formularios que debe rellenar con información relativa al titular de la reserva, la forma de pago, seguros contratados, y en su caso, la información de su tarjeta de crédito donde efectuar el cargo.

En este momento, el sistema debe avisar al cliente de la fecha en la que se incurriría en gastos por cancelación de la reserva. Si la fecha de reserva es próxima a la fecha de entrada, se establecerán restricciones en los métodos de pago, ofreciendo únicamente el pago inmediato con tarjeta de crédito.

El pago mediante transferencia o giro postal estará sujeto a unas condiciones que permitan la cancelación de la reserva por parte de la agencia de viajes sin incurrir en gastos de cancelación. Dado el caso en el que se aproxime la fecha en la que se incurra en gastos por cancelación y no se haya producido el pago



de la reserva, el sistema avisará al usuario de dicha situación. En caso de no recibir el pago, finalmente el sistema cancelará automáticamente dicha reserva y notificará al cliente esta acción.

En el caso de pago con tarjeta de crédito, se cargará inmediatamente el importe de la reserva a través de una pasarela de pago. En el caso de recibir un error en la validación de la tarjeta de crédito, se solicitará al cliente que revise sus datos y vuelva a introducirlos. Si tras varios intentos fallidos no se introduce una tarjeta válida, se procederá a finalizar el proceso de reserva. Si el pago se ha efectuado correctamente, se enviará una solicitud de reserva al proveedor pertinente y en caso de que éste confirme dicha reserva, el proceso finalizará con la notificación al cliente de reserva completada.

Si se diera el caso de no poder confirmar la reserva con el proveedor, se ejecutaría la devolución del importe de la reserva.

### *3.2. El modelo de datos*

Si en el anterior apartado hemos analizado las actividades y flujos que conforman el proceso de negocio, pasamos ahora a completar la especificación del sistema mediante la definición del modelo de datos que dará soporte a estas actividades.

El análisis detallado del proceso de negocio permite extraer las entidades de información que formarán parte de la entrada y salida de cada una de las tareas así como relaciones existentes entre estas entidades y sus multiplicidades.

Como en el caso anterior, mantener un modelo que ejerza de eje de comunicación sobre el cual analizar los requisitos del sistema ha resultado muy ventajoso en todas las etapas del desarrollo.

Puesto que el sistema a construir está muy condicionado por las estructuras y procesos definidos por los mayoristas, en la construcción del modelo de datos también se han tomado como *inputs* los modelos de datos asociados a los proveedores.

El diagrama del Apéndice B representa un ejemplo de un modelo de datos “espejo” de un proveedor concreto. Éste ha sido elaborado a partir de las estructuras de datos utilizadas en las operaciones definidas en el WSDL del servicio web.

## 4. Desarrollo de la aplicación

### 4.1 Metodología de desarrollo

Dada la estructura del equipo de desarrollo, en la que se integra un componente de la parte cliente, se ha considerado oportuno trabajar con una metodología de desarrollo en base a prototipos. Hablamos por tanto, de una metodología ágil.

Esta metodología define un proceso de desarrollo iterativo e incremental que facilita la comunicación entre las partes. En cada iteración, y cada nueva versión, se aumenta la funcionalidad y mejora en calidad respecto a la versión anterior.

*Ciclos cortos de desarrollo-entrega-evaluación.*

Mantener unos ciclos cortos de *elicitación de requisitos - desarrollo - entrega - evaluación* nos permite descubrir nuevos requisitos a todos los niveles que no quedaban definidos en la fase inicial, así como refinar la definición de aquellos ya establecidos.

Esta estrategia nos ha permitido llevar diferentes frentes abiertos a la vez. En este proyecto ha sido especialmente ventajoso ya que hemos tenido muchos problemas en la integración del principal proveedor, principalmente a la hora de sincronizar datos y completar las operaciones del proceso de reserva.

Al detectar errores en una fase temprana del prototipo ha sido posible avanzar en otro frente a la espera de la resolución de éstos por parte del proveedor.

Puesto que el programador de la parte cliente ha trabajado en la maquetación y el *frontend* de la aplicación, ha sido todavía mas sencillo entender cada parte del proceso definido, al poder asociar “pantallas” a tareas.

El desarrollo del proyecto ha estado dividido en 3 fases claramente diferenciadas:

1. Integración con proveedor principal e implementación de las operaciones requeridas para realizar consultas de disponibilidad: Una vez se ha alcanzado este objetivo, ha sido posible pasar el sistema a producción con una versión reducida del mismo. Durante este periodo, se han podido realizar reservas

vía telefónica a través de los agentes comerciales y los paneles de administración de los proveedores.

2. Adaptación del proceso de pre-reserva y reserva. A partir de este momento es posible la reserva de artículos a través de la plataforma online.
3. Creación de un backoffice donde la agencia pueda gestionar las reservas y ejecutar cuando se requiera las cancelaciones pertinentes.

El equipo de trabajo formado para llevar a cabo el proyecto ha estado compuesto por 3 personas. A continuación se enumerarán sus funciones y roles adoptados durante el transcurso del proyecto.

#### *Rol 1. Programador Frontend (Personal externo).*

La empresa contratante ha aportado recursos humanos como es el caso de un programador que la empresa tiene en plantilla.

Este programador ha adquirido el rol de desarrollador *frontend*, llevando a cabo el diseño, maquetación y programación de la parte cliente de la web.

No entraremos en detalles de las características de implementación de esta parte, pues queda fuera del alcance de este trabajo.

#### *Rol 2. Jefe de proyecto - Analista - Programador. (Personal interno).*

Además de desempeñar las funciones propias como jefe de proyecto, la persona que ha ocupado este puesto ha participado muy activamente en el análisis y diseño de la aplicación.

En cuanto al desarrollo de módulos, ha llevado a cabo la programación de la pasarela de pago y su integración en la aplicación.

#### *Rol 3. Analista programador. (Personal interno).*

Puesto que este es el rol que me ha sido asignado, se citan a continuación algunas de las funciones en las que he participado y en qué fase del proyecto se ha ejecutado cada una.

## Fase 1.

- Estudio de las estructuras de datos y operaciones de los servicios web de un proveedor externo para el posterior diseño e implementación de la base de datos asociada del mismo.
  - Análisis de información a integrar: corrección, estructuras, tipos.
  - Pruebas de operaciones y establecimiento de procesos a seguir. Argumentos de entrada y salida en operaciones. Flujos de llamadas para completar procesos (p.ej pre-reserva - gastos de cancelación – reserva)
  
- Diseño del modelo de datos “espejo” adaptado al del proveedor. (Apéndice B)
  - Creación del modelo lógico de base de datos
  
- Análisis y diseño del modelo de procesos:
  - Extracción de información de procesos en la agencia de viajes.
  - Reuniones periódicas con el proveedor.
  - Modelado BPMN incremental.
  
- Diseño del modelo de datos que da soporte a la aplicación
  - En base al modelo de datos creado en el paso anterior (modelo espejo), y modelos de datos de otros proveedores previamente estudiados se extraen patrones comunes y se adoptan en el nuevo modelo de datos de la aplicación.
  - Se incorporan al modelo de datos todas las entidades, atributos y relaciones necesarias para satisfacer la ejecución del proceso de negocio.
  - Modelo completo con soporte multi-proveedor.
  - Modelo completo con soporte a internacionalización.
  
- *Script* de sincronización entre información del proveedor y la base de datos “espejo”:
  - Programación de “script” PHP de sincronización entre la base de datos “espejo” del proveedor y los datos ofrecidos por el proveedor a través del servicio web.
  - Ejecución programada de consultas a las operaciones del servicio web del proveedor.
  - Filtrado y adecuación de información entrante.
  - Inserción / actualización en tablas locales

- Script de sincronización entre base de datos “espejo” y base de datos “roomsday”:
  - Programación de “script” de sincronización entre base de datos.
  - Actualización selectiva de información en base a modificaciones realizadas sobre datos locales, p.ej. las descripciones modificadas por la agencia de viajes de productos no deben actualizarse.

## Fase 2

- Programación del *motor de búsqueda multiproveedor*.
  - Subsistema con capacidad para lanzar en paralelo consultas de disponibilidad a distintos proveedores y a su vez, llamadas en paralelo con diferentes parámetros al mismo proveedor. PHP no dispone de una implementación que permita lanzar hilos de ejecución en paralelo. Se emula esta funcionalidad mediante llamadas a procesos hijos y tablas de base de datos como soporte para controlar la ejecución.
- Diseño y programación del subsistema de reserva *multiproveedor*
  - Subsistema con capacidad de ejecutar reservas a diferentes proveedores a través de una interfaz homogénea.
  - Desarrollo de un conjunto de clases que implementan la interfaz requerida por el proveedor para completar un proceso completo de consulta de disponibilidad, reserva o cancelación.

*A continuación se mostraran algunas partes del modelo de base de datos “roomsday” que cubren funcionalidades muy relevantes dentro del sistema.*

Ilustraremos mediante un ejemplo “patrón” como se ha diseñado el modelo de base de datos para las entidades del dominio con el fin de cumplir los requisitos establecidos de:

- *Internacionalización*. Cada entidad susceptible de ser traducida ha sido dividida en varias tablas, de modo que en una de ellas se mantengan los índices y atributos, y en otra las traducciones de los atributos a traducir.
- *Soporte multi -proveedor*. Una tabla intermedia alberga los índices referentes a la ciudad del sistema y a la ciudad del proveedor. En esta asociación se guardan además datos serializados que cada proveedor puede aportar con una estructura propia.

- *Información editable.* Todos aquellos campos susceptibles de ser editados por la agencia de viajes van acompañados por *flags* que indicarán al sincronizador si debe sobrescribir el contenido o dejarlo tal y como está. Latitud - Longitud con los flags *latitud\_edit* y *longitud\_edit*

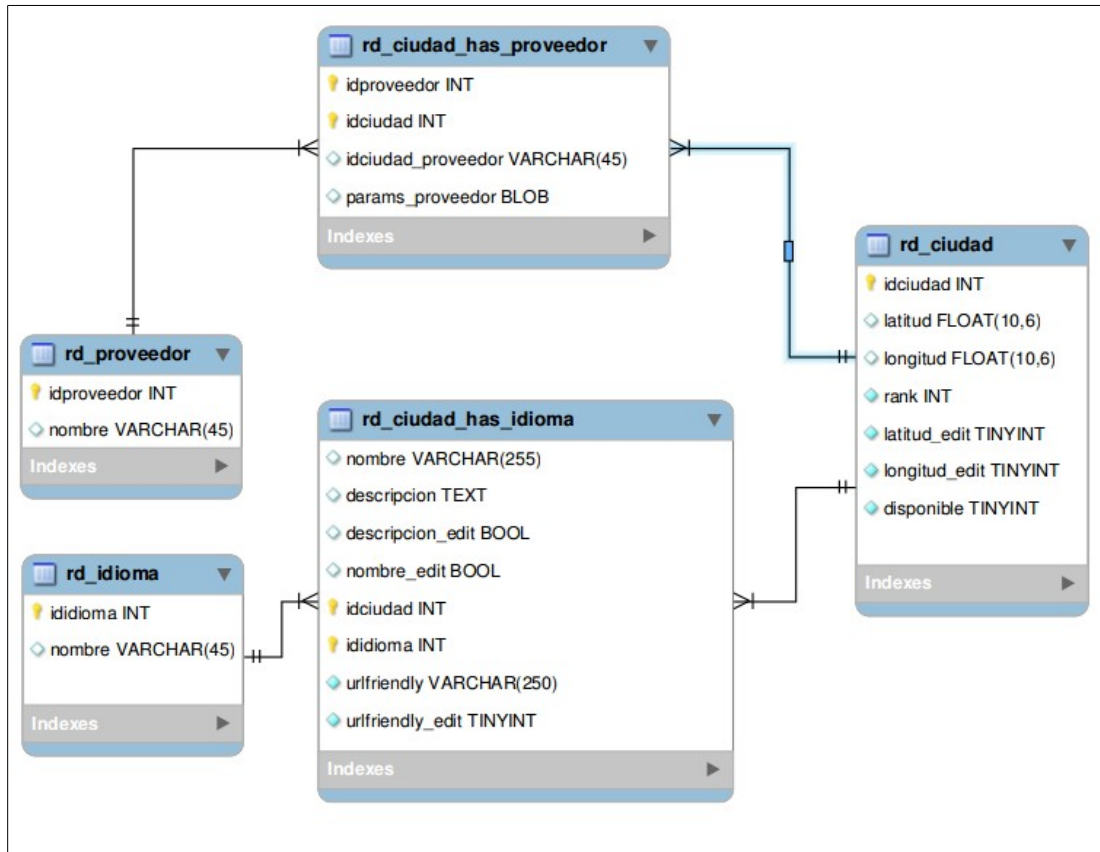


Ilustración 1: Ejemplo patrón multi-idioma y multi-proveedor

También podemos observar como se ha dado soporte a diferentes subsistemas dentro del modelo de base datos, por ejemplo:

- *Subsistema de búsquedas:* Además de persistir las búsquedas, este subsistema hace uso de tablas auxiliares que controlan el flujo de resultados recibidos desde los diferentes proveedores (\*timeout), así como de tablas de resultados agregados que facilitan la consulta de datos calculados.

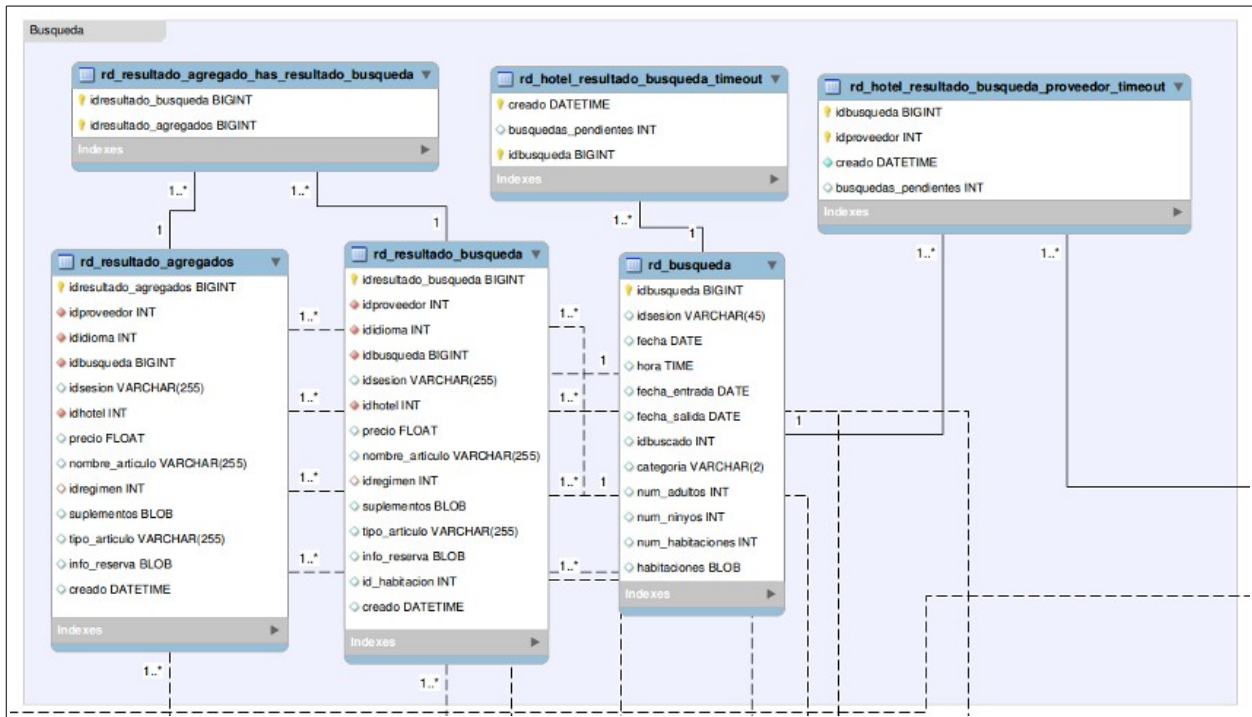


Ilustración 2: Subsistema de búsqueda

### Subsistema de reserva

- Estas tablas dan persistencia a los datos necesarios para efectuar pre-reservas y reservas. De igual modo, almacenan los gastos de cancelación asociados a éstas y las transacciones realizadas para su confirmación.

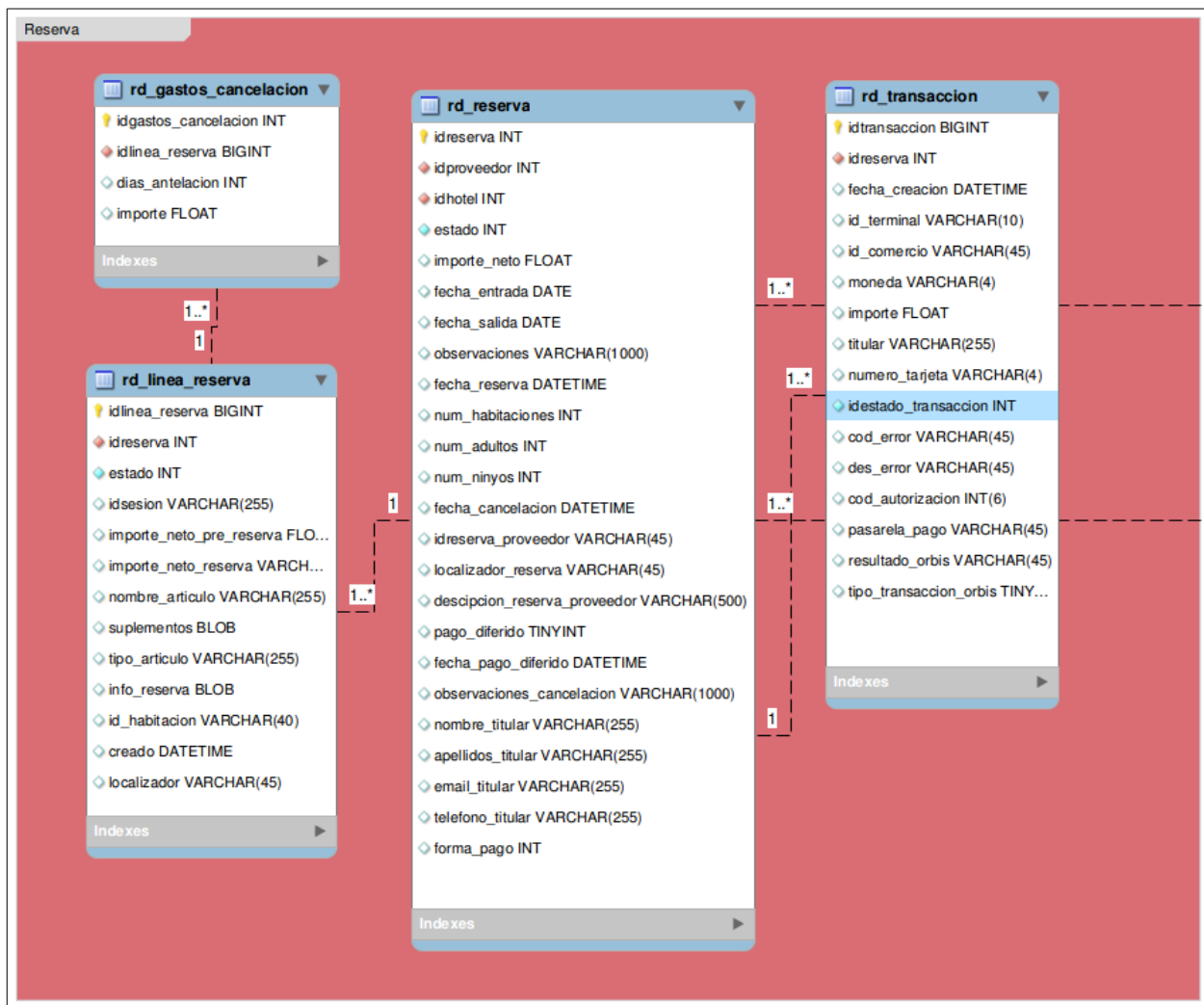


Ilustración 3: Subsistema de reserva



## 4.2 Entorno de desarrollo

### 4.2.1 Tecnología

Las tecnologías utilizadas en el desarrollo del proyecto así como la arquitectura de la aplicación han venido condicionadas por parte del cliente, ya que este proyecto reutiliza ciertas clases y código HTML de otro proyecto similar ya en funcionamiento. Cabe destacar que las posteriores ampliaciones y mantenimiento de la aplicación serán llevadas a cabo por el propio equipo técnico del cliente.

El proyecto se ha desarrollado utilizando una combinación de tecnologías más que probada y que ha demostrado su sencillez, potencia y versatilidad. El lenguaje de programación utilizado en la parte servidora es PHP. Concretamente, en su versión 5.2, ejecutado sobre un servidor Apache. PHP, al ser un lenguaje interpretado minimiza la infraestructura necesaria para ejecutarse y hace el proceso de desarrollo muy ágil; puede ser ejecutado bajo un gran número de servidores sin necesidad de compilación y su curva de aprendizaje es relativamente corta. Cabe destacar que a pesar de ser un lenguaje muy completo en lo referente a funcionalidades ofrecidas, se ha tenido que buscar una solución alternativa en el momento en que se ha requerido de ejecución multi-hilo, ya que PHP no la ofrece de forma nativa.

El sistema gestor de base de datos elegido para dar soporte a la aplicación es MySQL y el acceso a base de datos se ha gestionado a través de la extensión de PHP PDO.

PDO<sup>1</sup> proporciona una capa de abstracción de *acceso a datos*, lo que significa que, independientemente de la base de datos que se esté utilizando, se usan las mismas funciones para realizar consultas y obtener datos. PDO no proporciona una abstracción de bases de datos; no reescribe SQL ni emula características ausentes.

En esta primera fase se ha optado por prescindir, siempre que ha sido posible, del uso de frameworks y componentes de terceras partes, ya que en fases posteriores se pretende llevar a cabo la implantación del compilador “hiphop” utilizado por facebook. Este compilador limita el uso de ciertas características que acompañan a las últimas versiones de PHP, en concreto, a partir de la versión 5.3.

---

<sup>1</sup><http://www.php.net/manual/es/intro.pdo.php>

*Hiphop*<sup>1</sup> es una herramienta de compilación de código PHP que nació en el departamento de investigación de facebook con el fin de optimizar la ejecución de la aplicación y que desde entonces se utiliza en producción de forma exitosa. Este compilador de PHP, realiza un análisis estático de código y lo compila a C++ optimizado, incrementando la velocidad de ejecución y reduciendo el tiempo de CPU.

El proyecto está alojado en un repositorio gestionado a través de la herramienta de control de versiones Subversion.

### 4.3 Arquitectura de la aplicación.

Como hemos citado anteriormente, el entorno de trabajo ha venido condicionado por un proyecto previo aportado por el cliente.

De este marco de trabajo podemos destacar:

- Utilización del fichero index.php como único punto de entrada a la aplicación donde se produce la inicialización o bootstrap de diferentes configuraciones de la aplicación.
  - Creación de la conexión a la base de datos.
  - Creación de instancia de un logger
  - Creación de rutas.
  - *Dispatcher*
  - *Autoloader*
  - Filtrado de parámetros
- Uso extensivo de directivas de apache a través de htaccess<sup>2</sup>:
  - Reescritura URL
    - mod\_rewrite
  - Caché
    - mod\_headers
  - Compresión y optimización
    - mod\_deflate

---

1 <https://github.com/facebook/hiphop-php/wiki>

2 Un fichero .htaccess (hypertext access), también conocido como archivo de configuración distribuida, es un fichero especial, popularizado por el Servidor HTTP Apache que permite definir diferentes directivas de configuración para cada directorio (con sus respectivos subdirectorios) sin necesidad de editar el archivo de configuración principal de Apache. [Wikipedia]

- No existe ningún patrón estructural definido al uso. El código se ha estructurado en base a vistas o módulos. Entiéndase por módulo o vista una página completa generada tras una llamada al fichero php correspondiente. Cada módulo contiene directorios que albergan las hojas de estilo y scripts que se ejecutan en el lado cliente. El propio fichero php ejecuta métodos de acceso a datos y a su vez genera el HTML requerido para mostrarlos.

Se ha generado una capa de abstracción de operaciones de los servicios web a partir de los ficheros WSDL expuestos por los proveedores mediante la herramienta *wsdltophp*<sup>1</sup>. Esta herramienta permite generar un conjunto de clases que implementan todas las operaciones definidas por el servicio web. Estas clases hacen uso del cliente SOAP nativo de PHP, y se encargan de facilitar entre otras, la configuración y construcción de dicho cliente, la ejecución de operaciones y captura de excepciones.

---

<sup>1</sup> [www.wsdltophp.com](http://www.wsdltophp.com)

## 5. Demostración de la aplicación

Para finalizar, comentaremos algunas secciones relevantes de la plataforma, ya accesible online a través de [www.roomsday.com](http://www.roomsday.com).

### 5.1 Buscador

En primer lugar, el buscador solicita los parámetros mínimos para poder lanzar una búsqueda a los proveedores. Como vemos en la imagen de la derecha, se ha incorporado un campo con función *autocompletar* para sugerir resultados de forma asíncrona y agilizar las búsquedas. Los resultados sugeridos incluyen ciudades, hoteles y marcas. En este caso, marcas geográficas.

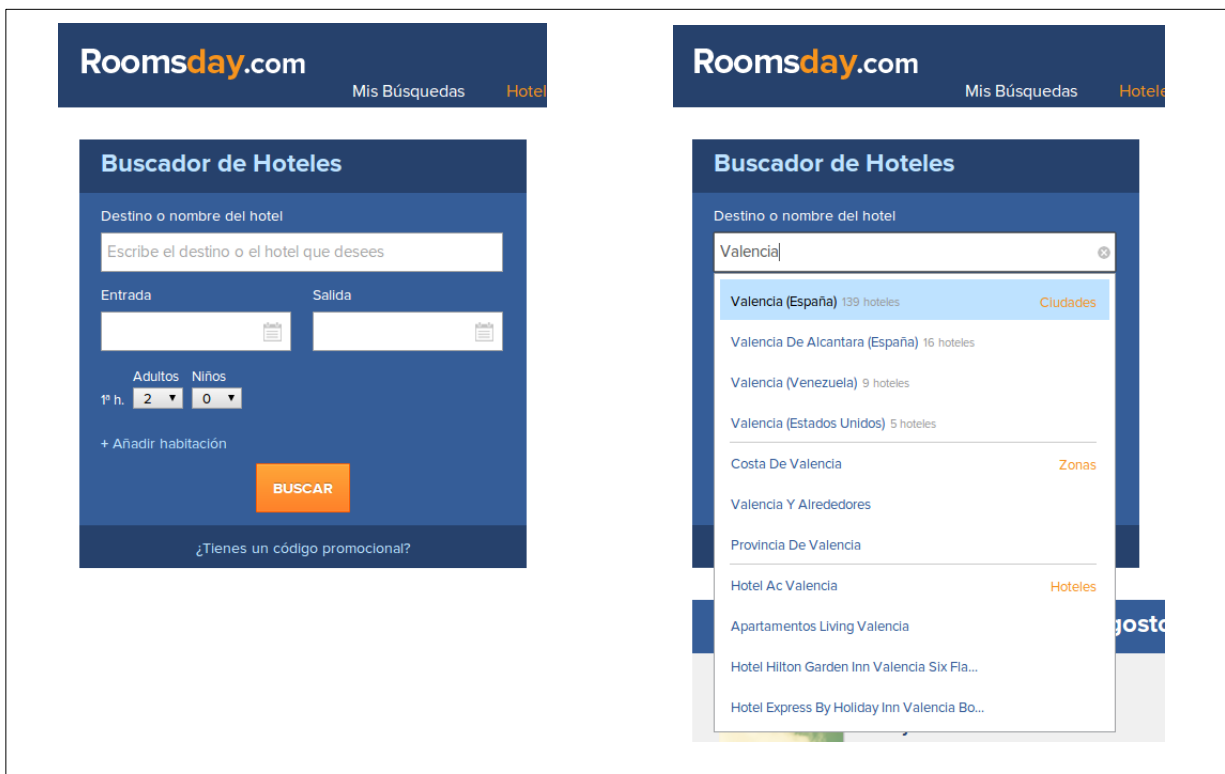


Ilustración 4: Buscador / Función autocompletar

### 5.3 Listado de artículos disponibles

Una vez ejecutada la búsqueda de disponibilidad con los parámetros obtenidos en el paso anterior, mostramos al usuario la ficha del hotel con los mejores precios obtenidos. El buscador de la parte izquierda ofrece las mismas funcionalidades que el visto en el punto anterior, pero en la siguiente fase se acompañará de filtros que permitan afinar la búsqueda por distintos parámetros. Actualmente el orden de aparición de los resultados viene dado por el número de reseñas positivas en diferentes rankings de hoteles.

**Roomsday.com** Atención al cliente **962 84 84 84**  
 contacto@roomsday.com

Mis Búsquedas Hoteles Escapadas Blog

## 10 hoteles en Playa De Gandia (España)

### Buscador de Hoteles

Destino o nombre del hotel  
**Playa De Gandia (España)**

Entrada: 27/09/2013 Salida: 29/09/2013

Habitaciones: 1

Adultos Niños: 2 0

**BUSCAR**

¿Tienes un código promocional?

2 Noches, 2 Adultos (1 hab.) Código promocional

### Hotel Gandia Playa ★★★

C/Devesa, 17 - 46730 Playa De Gandia - España

Situado a 150m. de la playa de Gandia. Dispone de 126 habitaciones distribuidas en 9 plantas. El hotel cuenta con salón comedor tipo buffet, 2 piscinas (para adultos y niños).

68 Fav

**¡SIN SORPRESAS!**  
 Precio total para 2 noches

Doble Tarifas 2013 Netos	Sólo alojamiento	100€	Reservar
Doble Tarifas 2013 Netos	Alojamiento y desayuno	104€	Reservar
Doble Tarifas 2013 Netos	Media pensión	115€	Reservar
Doble Tarifas 2013 Netos	Pensión completa	123€	Reservar

### Hotel Tres Anclas ★★★

C/valldigna, Nº 11 - 46730 Playa De Gandia - España

Situado a tan solo 100 metros de las Playa, al final del Paseo Marítimo. Este hotel de playa cuenta con 10 plantas y un total de 437 habitaciones que son dobles, superior, estudios y

47 Fav

Ilustración 5 Listado de resultados

## 5.4 Página de reserva ( 1 )

Una vez se ha llegado a esta página, se efectúa la pre-reserva del artículo seleccionado y en los formularios de la columna izquierda se solicitan los datos del titular de la reserva y ocupantes de las habitaciones. En la columna de la derecha, se puede visualizar un resumen del artículo a reservar y un enlace a gastos por cancelación.

**Roomsday.com** Mis Búsquedas <sup>2</sup> Hoteles Escapadas Blog Atención al cliente  
**962 84 84 84**  
contacto@roomsday.com

### Datos del Cliente

Nombre y Apellidos

Usuario  ❌

Dirección

Dirección  ✔️

Código Postal Ciudad

Código Postal  Ciudad  ❌

Correo electrónico

Correo electrónico  ❌

Confirme correo electrónico

Confirme correo electrónico  ❌

Número de Teléfono

Número de Teléfono  ❌

### Información para el hotel

Nombre y Apellidos **1er Adulto**

Nombre y Apellidos  ❌

Nombre y Apellidos **2º Adulto**

Nombre y Apellidos  ✔️

Observaciones

Observaciones  ❌

### Tu Reserva

**Hotel Gandia Playa \*\*\***

**Entrada:** Miércoles, 30 de Noviembre de 2013

**Salida:** Domingo, 12 de Diciembre de 2013

**Noches:** 8

**1ª Habitación**

Habitación estándar doble (2 Adultos)

**Régimen:** Alojamiento y Desayuno

**Precio final: 423,40€**

¿Necesitas Ayuda?  
**962 84 84 84** [Política de cancelación](#)

Ilustración 6: Página de reserva (1)

## 5.5 Página de reserva ( II )

Para finalizar el proceso de reserva, el usuario debe elegir una forma de pago y los datos requeridos para poder efectuarlo. El sistema ofrecerá las modalidades de pago posibles en base a la fecha en la que se incurra en gastos por cancelación.

### Seguro de Viaje

Seguro de asistencia en viaje 22,90€ [¿Qué es?](#)

Asistencia y anulación (Recomendado) 18,93€ [¿Qué es?](#)

No quiero contratar seguro

### Condiciones de pago

Quiero pagar ahora el 100% de la reserva (423,40€)

Quiero pagar ahora 10€, y el resto antes del 15/06/2013

El importe restante deberá pagarse antes de los 30 días de la fecha de salida. Para ello, te enviaremos un e-mail para que accediendo a la web puedas realizar el pago pendiente. Para más dudas puedes consultar nuestra política de cobros.

### Forma de pago

Tarjeta de Crédito

Transferencia / Giro Postal

He leído y acepto las políticas de privacidad, condiciones generales, política de cobros, y condiciones de cancelación de ROOMSDAY para esta reserva y acepto el pago de 10,00€ .

### Datos de la tarjeta

Número de tarjeta

Titular de la tarjeta

Fecha Caducidad CVV [¿Qué es?](#)

# Roomsdays.com

¿Quiénes somos?  
© 2013 Roomsdays Travel S.L.

[f](#) [t](#) [g+](#)

Formas de pago  
Política de privacidad  
Preguntas Frecuentes FAQ

Ilustración 7 Página de reserva (2)

## 6. Conclusiones

En el momento de la entrega de este documento, todavía restan funcionalidades por implementar y la revisión de algunas de ellas. Sin embargo, el trabajo realizado permite extraer conclusiones en diferentes ámbitos.

En el ámbito de la captura de requisitos, podemos argumentar en favor del sistema utilizado, ya que ha facilitado la comunicación entre las partes y ha generado dudas y nuevos requisitos que no aparecían en las primeras especificaciones textuales. De cara a la integración con un sistema fuertemente ligado a un proceso de negocio, como es el que nos atañe, ha servido como hoja de ruta para determinar qué partes del sistema propio interactúan con los proveedores.

En cuanto a la metodología utilizada, cabe decir que ha sido de ayuda poder ir contrastando de forma incremental los prototipos desarrollados. Sin embargo, considero que este sistema también tiene ciertos inconvenientes que se van descubriendo a medida que se evolucionan los prototipos. Se observa que un error de diseño en las primeras iteraciones puede generar mucho trabajo de refactorización de código y rediseño de base de datos a medio y largo plazo. Si no se toman acciones para arreglar estos errores de diseño en las primeras fases del prototipo, éste puede quedar condenado a arrastrar errores durante toda su vida útil pudiendo generar además problemas en otras partes del sistema.

Como hemos comentado en el apartado de “Tecnología”, se han impuesto desde el cliente ciertos requisitos al respecto. Destacar que desde la empresa no hemos compartido algunos de estos requisitos. Por ejemplo, el de “optimización temprana”, condicionando el desarrollo de la aplicación a utilizar versiones ya obsoletas en pro de utilizar en un futuro un sistema de optimización muy poco desarrollado. Consideramos que existen otras formas de optimizar una aplicación PHP más extendidas, probadas y desde nuestro punto de vista, más eficaces dados los requisitos de la aplicación: sistemas de caché, técnicas de optimización de código, compresión de código, reducción de accesos a bases de datos, optimización de bases de datos y consultas, etc.

Por otra parte, consideramos poco eficiente desarrollar *desde cero* una aplicación que tiene vistas a crecer en número de funcionalidades y usuarios. Existen disponibles multitud de *frameworks* de probada solvencia que agilizan el desarrollo de aplicaciones, a la vez que estandarizan el modelo de programación, haciendo



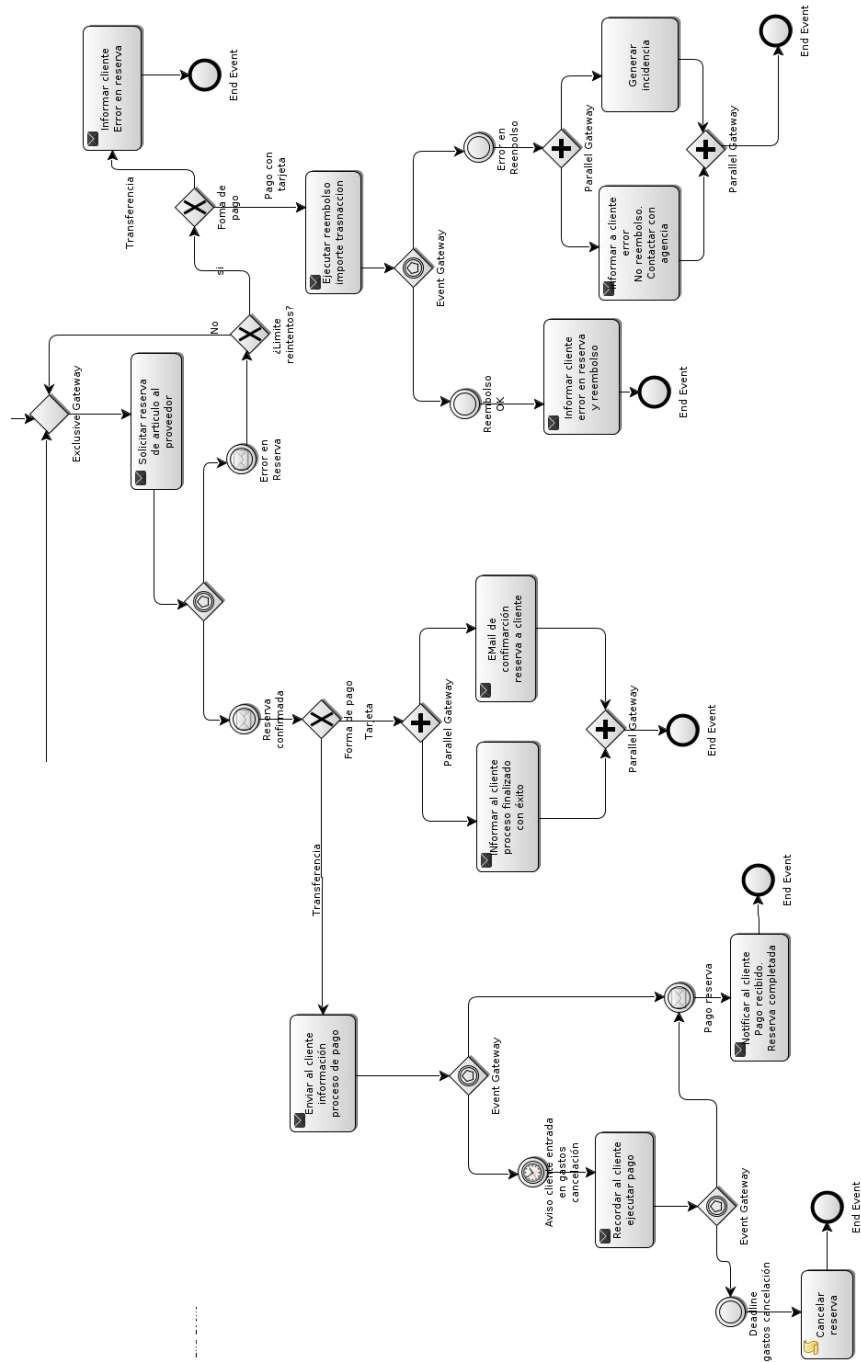
posible la incorporación de nuevos miembros al equipo de desarrollo con una curva de aprendizaje relativamente corta.

Estos frameworks han sido concebidos utilizando buenas prácticas de desarrollo orientado a objetos y entre otras, han pasado pruebas de seguridad, funcionales y de rendimiento.

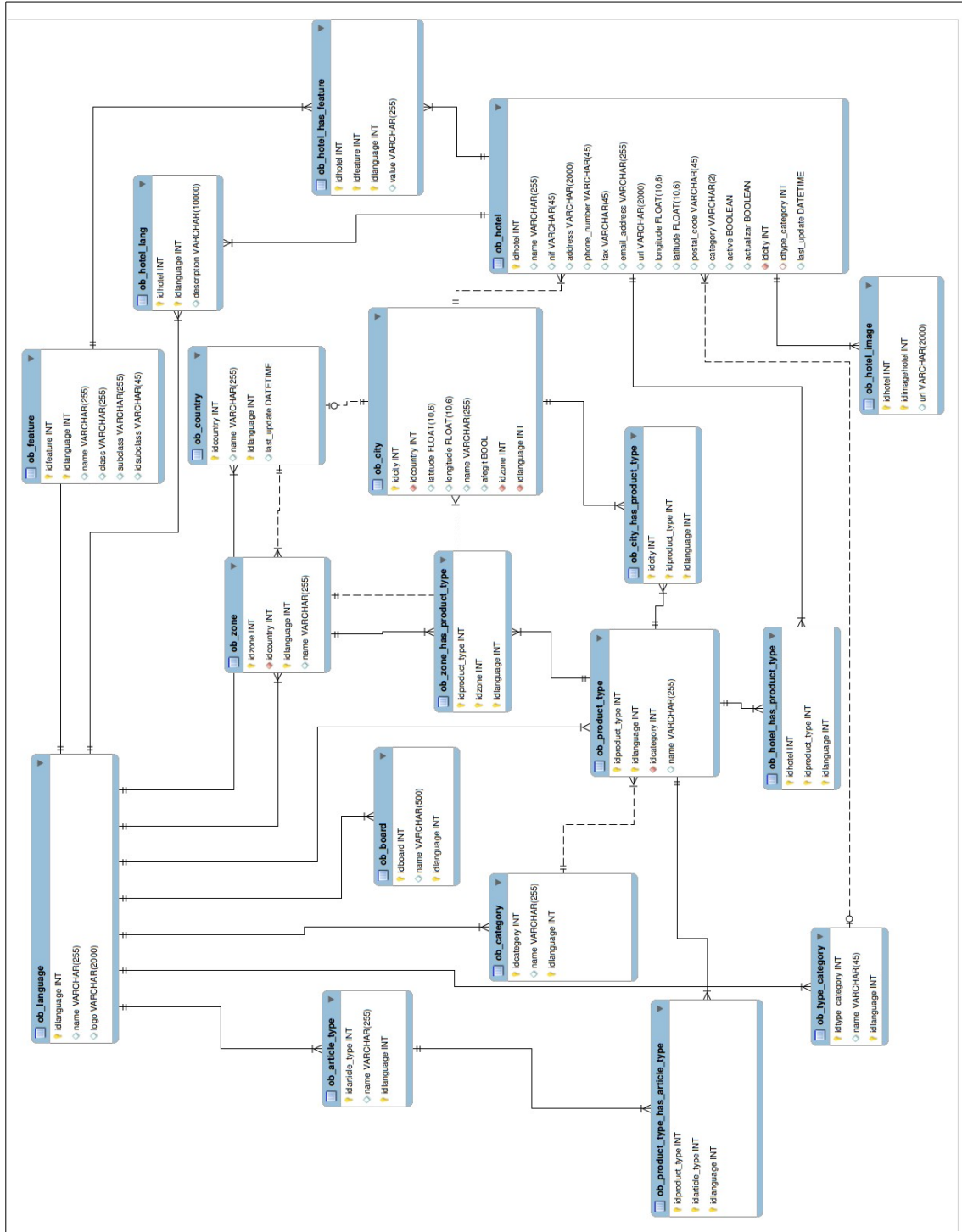
Por todo esto, no consideramos una buena práctica anteponer la velocidad de ejecución a la seguridad, escalabilidad y mantenibilidad de una aplicación. Si bien entendemos que la optimización es una característica a tener en cuenta y que debe aplicarse a todos los niveles del sistema, consideramos que primero deben quedar cubiertas unas necesidades básicas que no comprometan el desarrollo del proyecto.



# Apéndice A. Modelo de procesos de negocio (II)



Apéndice B. Modelo de datos del proveedor.



## Bibliografía

PDO: <http://php.net/manual/es/book.pdo.php>

HIPHOP: <http://corporate.tuenti.com/es/dev/blog/the-epic-tale-of-moving-to-hiphop>

HIPHOP: <https://github.com/facebook/hiphop-php>

Yaoqiang: <http://sourceforge.net/projects/bpmn/>

Htaccess: <http://es.wikipedia.org/wiki/Htaccess>

WSDL to PHP: <http://www.wsdltophp.com/>