



**Universidad**  
Zaragoza

## Proyecto Fin de Carrera

Diseño e implementación de prototipo *front end*  
con estructura adaptativa para un portal de reserva  
de hoteles

Autor

Pablo Gil Torres

Director

Chris Anderson

Ponente

José Javier Merseguer

Escuela de Ingeniería y Arquitectura

2014



## Agradecimientos

Quiero agradecer a mi familia su apoyo y comprensión durante este largo proceso.

A José Merseguer por su apoyo, motivación y paciencia durante estos seis años.

Y a Chris Anderson y BoomWorks por hacerme parte de su equipo y brindarme la oportunidad de llevar a cabo este proyecto.



# Diseño e implementación de prototipo *front end* con estructura adaptativa para un portal de reserva de hoteles

## Resumen

Este proyecto es producto de la respuesta de la empresa BoomWorks a las necesidades de la empresa Youth Hostelling Asociation Australia (YHA Australia). Estas necesidades son un rediseño de su web móvil para Australia y la creación de portales con la capacidad de realizar reservas para países del continente asiático.

Ya que ambos proyectos eran similares en términos de funcionalidad y requisitos se decidió aunar los esfuerzos bajo una misma solución: Implementar un sistema basado en el diseño adaptativo con las mismas características y contenido a través de dispositivos.

Debido a lo novel de la metodología de *responsive design* en 2012 y la falta de experiencia de BoomWorks con la misma en proyectos de larga escala, se decidió gestionar la parte de estructura, estilo y comportamiento de interacción de manera aislada y controlada, en un prototipo PHP.

Este prototipo debía habilitar las siguientes funciones:

- Buscar un hotel.
- Inspeccionar habitaciones y tarifas en el hotel seleccionado.
- Seleccionar habitación o cama.
- Comprobar disponibilidad.
- Llevar a cabo la reserva.

Los detalles concretos del sistema de plantillas incluyen:

- Campo de búsqueda inteligente por geolocalización, sugerencias o cadenas de texto.
- Utilidad dinámica de calendario para comprobar disponibilidad de fechas.
- Gestión de mapas y localización de resultados.
- Validación de formulario y habilitación de elementos de formulario HTML5
- Compatibilidad con navegadores con soporte limitado.
- Gestión de imágenes en entornos con resoluciones reducidas.



## Índice de contenidos

Resumen .....	I
Introducción.....	1
1. Motivación y contexto del proyecto .....	1
Diseño .....	5
1. Generación de requisitos y estrategia de implementación <i>front end</i> .....	5
2. Diseño de la arquitectura .....	7
2.1 Páginas y módulos del prototipo .....	7
2.3 Estructura de la arquitectura de página.....	11
2.4 Gestión de resoluciones de dispositivo.....	13
Implementación .....	15
1. Campo de búsqueda.....	15
2. Calendario.....	16
3. Gestión de mapas.....	18
4. Validación de formularios de la fase de reserva.....	20
5. <i>Graceful degradation</i> .....	22
6. Gestión de imágenes .....	24
7. Retos y problemas .....	25
Pruebas del sistema.....	27
1. Estrategia .....	27
2. Entorno de pruebas .....	27
3. Roles .....	27
4. Tipos de pruebas .....	28
5. Problemas destacados .....	28
5.1 Comportamiento errático de la pantalla al interactuar con elementos de tipo input en Android 2.2.....	28
5.2 Google Maps imposibilita la navegación de la pantalla en entornos móviles..	29
5.3 Funcionalidad de restaurado de fechas en el calendario.....	29
5.4 Aspect ratio en los carruseles de imágenes .....	30
Conclusiones.....	32
1. Resultados obtenidos .....	32
2. Lecciones aprendidas.....	32
3. Líneas de trabajo futuras.....	33
Anexos.....	35
Anexo A: Investigación de competidores móviles .....	35
Anexo B: Bocetos de <i>wireframes</i> móviles .....	40

B1 Resultados de búsqueda .....	40
B2 Herramienta de filtrado .....	41
B3 Herramienta de búsqueda.....	42
B4 Detalles de hotel.....	43
B5 Calendario .....	44
B6 Herramienta de filtrado .....	45
Anexo C: <i>Wireframes</i> móviles .....	46
C1 <i>Map browse</i> .....	46
C2 <i>String search</i> .....	47
C3 <i>Calendar</i> .....	48
C4 <i>Hostel details</i> .....	49
C5 <i>Select rooms and guests</i> .....	50
C6 <i>Payment</i> .....	51
C7 <i>Confirmation</i> .....	52
Anexo D: <i>Wireframes</i> de escritorio .....	53
D1 <i>Home</i> .....	53
D2 <i>Map Search</i> .....	53
D3 <i>String Search</i> .....	54
D4 <i>Hostel details</i> .....	54
D5 <i>Select Rooms</i> .....	55
D6 <i>Select guests</i> .....	55
D7 <i>Payment</i> .....	56
D8 <i>Confirmation</i> .....	56
Anexo E: Diseños visuales móviles .....	57
E1 Página de inicio .....	57
E2 Resultados de búsqueda por localización.....	58
E3 Resultados de búsqueda filtrados por disponibilidad.....	59
E4 Calendario .....	60
E5 Detalles de hotel.....	61
E6 Selección de habitación.....	62
E7 Selección de huéspedes .....	63
E8 Detalles de huésped y pago .....	64
E9 Confirmación de la reserva .....	65
Anexo F: Ejemplos visuales de resolución de escritorio y lenguaje japonés .....	66
F1 Página de inicio .....	66
F2 Resultados de búsqueda por localización en lenguaje japonés .....	67
Anexo G: Distribución de horas de trabajo mensuales.....	68



Referencias .....	70
-------------------	----

## Índice de figuras

<i>Figura 1 Esquema de la estructura de la página .....</i>	<i>11</i>
<i>Figura 2 Ejemplo de estilos extendidos a través de la anchura de la página.....</i>	<i>12</i>
<i>Figura 3 Cabecera de escritorio .....</i>	<i>12</i>
<i>Figura 4 Cabecera móvil.....</i>	<i>12</i>
<i>Figura 5 Campo de búsqueda .....</i>	<i>15</i>
<i>Figura 6 Calendario de selección de fechas .....</i>	<i>16</i>
<i>Figura 7 Mapa de detalles de hotel.....</i>	<i>18</i>
<i>Figura 8 Mapa de resultados de búsqueda .....</i>	<i>19</i>
<i>Figura 9 Detalle del formulario de pago .....</i>	<i>20</i>
<i>Figura 10 Ejemplo de validación de formulario no superada .....</i>	<i>21</i>
<i>Figura 11 Carrusel de página de inicio redimensionado para entornos móviles.....</i>	<i>24</i>
<i>Figura 12 Capa de protección para la interacción con Google Maps .....</i>	<i>29</i>

## Índice de código

<i>Código 1 ejemplo de paso de parámetros a la funcionalidad de módulo .....</i>	<i>9</i>
<i>Código 2 Estructura de la implementación del patrón de diseño módulo .....</i>	<i>10</i>
<i>Código 3 Ejemplo de break point para resolución de 1024px especificado con media queries .....</i>	<i>13</i>
<i>Código 4 Ejemplo de filtro propietario de Microsoft para implementar transparencia en Internet Explorer .....</i>	<i>22</i>
<i>Código 5 Inline-block hack para dotar comportamiento inline-block a elementos en Internet Explorer 7 .....</i>	<i>22</i>
<i>Código 6 Ejemplo de hojas de estilo cargadas condicionalmente basadas en versiones de Internet Explorer.....</i>	<i>23</i>
<i>Código 7 Demostración de la reescritura dinámica de la referencia a imágenes para adaptarlas a la resolución del dispositivo.....</i>	<i>24</i>



# Introducción

## 1. Motivación y contexto del proyecto

Youth Hostelling Asociation Australia (YHA Australia) [1] es una asociación sin ánimo de lucro miembro del consorcio internacional HI (*Hostelling International*) [2] cuya actividad principal es gestionar la mayor red australiana de hoteles y alojamiento de bajo presupuesto.

En 2009 BoomWorks y YHA se asociaron para desarrollar una versión móvil de su web. El iPhone había logrado introducirse en el mercado local de manera considerable y las investigaciones contextuales en los propios hoteles indicaban que el número de viajeros en posesión de *smartphones* era suficiente para justificar la estrategia móvil. El requisito principal de dicha web era permitir que los usuarios realizasen reservas de hoteles desde sus dispositivos móviles.

Durante el último lustro el número y diversidad de dispositivos móviles disponibles en el mercado ha crecido exponencialmente. Los fabricantes de estos productos intentan diferenciarse y abarcar cuota de mercado desarrollando dispositivos cuyos formatos, capacidades técnicas, líneas de producto y segmentos de mercado objetivo sean capaces de desmarcarse de sus competidores. Estas estrategias de mercado se rigen por una serie de condiciones tecnológicas y de negocio que dan como resultado una gran variedad de características a lo largo del espectro de productos disponibles. Un elemento en particular, la pantalla, sufre una gran variación de tamaños, relaciones de aspecto, resoluciones y densidad de píxeles.

El acceso a Internet es una característica predominante en los dispositivos móviles y por lo tanto una de las principales actividades de usuario es visitar contenido web. Tradicionalmente este tipo de contenido ha sido diseñado y desarrollado para ser consumido en equipos de sobremesa, con las resoluciones y tamaños de pantalla de los monitores asociados.

Sin embargo, el acceso a dichos contenidos desde dispositivos móviles deja patente una serie de problemas tanto de usabilidad y accesibilidad [3][4] como de consumo de datos.

Las soluciones propuestas para alcanzar un grado de visualización y utilización aceptable en varias resoluciones han sido diversas. Los fabricantes de productos han dotado a los dispositivos de mayor densidad de píxeles y de herramientas de zoom. Los ingenieros web han desarrollado una serie de alternativas y mecanismos para presentar de manera apropiada los contenidos: *Feature sniffing*, detección de dispositivo, componentes *server-side*, etc. Las distintas soluciones resuelven el problema bajo determinadas circunstancias pero carecen de robustez y escalabilidad.

Ethan Marcotte sentó, aún sin saberlo, lo que serían las bases de su celebrado concepto de desarrollo basado en *responsive design* en 2009 en la prestigiosa publicación online *A List Apart* [5]. En este artículo Ethan proponía huir de restricciones de anchura fija a través de un sistema de rejillas fluidas y definía el concepto de diseño fluido [6].

Un año más tarde y en *A List Apart* de nuevo, Marcotte publicaba un artículo titulado “*Responsive Web Design*” [7] donde ampliaba el concepto de diseño fluido y le daba un cariz estratégico. Ya no se trataba de aprovechar la resolución del dispositivo sino de presentar el contenido de la manera correcta para el entorno en cuestión: a la estructura fluida se sumaba como elemento fundamental de esta propuesta las *media queries*.

Las *media queries* [8] son capaces de ejecutar bloques de propiedades CSS [9] de manera condicional, tomando como parámetros elementos relativos al dispositivo utilizado. Ahora la página no solo podía ser optimizada para resoluciones dadas sino que un mismo sistema podía ser diseñado de diferentes maneras para presentar contenido y funcionalidad de la manera óptima para cada resolución.

A finales del 2012 la tecnología en el ámbito del *smartphone* había evolucionado considerablemente. Las páginas basadas en *responsive design* empezaban a hacerse más habituales, siendo adoptadas por firmas conocidas y apareciendo en publicaciones especializadas del sector.

A través de otros proyectos realizados con BoomWorks, YHA disponía de una estructura de gestión de contenidos web más versátil y adaptable.

En una serie de reuniones de cariz estratégico se propuso el rediseño de la web móvil de YHA (<http://m.yha.com.au/>) para darle un aspecto más moderno, servir el contenido de una manera más eficaz y dedicada a las diferentes resoluciones y aprovechar al máximo las capacidades de la nueva infraestructura web.

YHA posee también una serie de versiones de su página web traducidas a otros idiomas (YHA *multilingual*):

- YHA Australia en chino: <http://cn.yha.com.au/>
- YHA Australia en japonés: <http://jp.yha.com.au/>
- YHA Australia en coreano: <http://kr.yha.com.au/>

Dichas versiones tradicionalmente han sido un subconjunto de la web principal con un mínimo de contenido y funcionalidad presente. La razón, evitar costes innecesarios de traducción y mantenimiento.

Las páginas YHA *multilingual* presentaban una experiencia de usuario pobre, ya que quedaba patente que sólo una pequeña parte de la web original había sido traducida.

El alcance o ámbito principal del proyecto de rediseño móvil seguía siendo proporcionar la capacidad de hacer reservas de hoteles. Dicha funcionalidad coincidía con las necesidades de las páginas YHA *multilingual*. BoomWorks propuso combinar los dos proyectos bajo una misma solución: Implementar un sistema basado en el diseño adaptativo con las mismas características y contenido a través de dispositivos.

La estrategia a seguir también dotaba a las páginas multi-lenguaje de presencia móvil por derecho propio. La estructura de las diferentes presencias web de la marca YHA era pues la siguiente:

- YHA para países de habla no inglesa, cualquier resolución: Página web con estructura basada en *responsive design* y funcionalidad limitada, traducida al lenguaje pertinente.

- YHA en inglés para tabletas y móviles: Versión de la página web anteriormente descrita, traducida al inglés.
- YHA en inglés para escritorio: Portal web completo, rediseñado al margen de este proyecto.

La aplicación permite al usuario realizar las siguientes tareas:

- Buscar un hotel.
- Inspeccionar habitaciones y tarifas en el hotel seleccionado.
- Seleccionar habitación o cama.
- Comprobar disponibilidad.
- Llevar a cabo la reserva.





# Diseño

## 1. Generación de requisitos y estrategia de implementación *front end*

La metodología adoptada por BoomWorks es el diseño iterativo centrado en el usuario [10]. Para el proyecto con YHA se realizó una ronda de análisis de negocio (*business analysis*) que consistió en entrevistas con los responsables del negocio, análisis de competidores y testeo de guerrilla con usuarios para validar los conceptos iniciales. A través de estas actividades preliminares se obtuvo una serie de requisitos de alto nivel y resultados deseados (*outcomes*) que fijaron la dirección del diseño.

Una vez completado dicho diseño, y tras la consiguiente validación vía testeo de usuarios, se genera una serie de requisitos más concretos que describen las necesidades de la implementación.

Requisitos funcionales:

- Buscar hoteles: El sistema permitirá al usuario realizar una búsqueda de hoteles por palabras clave sugeridas, geolocalización y entrada libre de texto. También permitirá buscar hoteles mediante selección de zona geográfica a través de un mapa.
- Seleccionar fechas: El sistema permitirá al usuario elegir las fechas de su estancia.
- Visualización de detalles de hotel: El sistema presentará al usuario una página con los detalles del hotel seleccionado. Dichos detalles incluirán la posición del hotel en un mapa de coordenadas, las características del alojamiento y puntuación y reseñas de viajeros.
- Selección de habitación: El sistema presentará al usuario las habitaciones disponibles basadas en las fechas introducidas y las características de las instalaciones. El sistema permitirá seleccionar una habitación al usuario.
- Introducción de número de huéspedes: El sistema permitirá introducir el número de huéspedes. El sistema permitirá introducir dichos datos en las categorías de miembro y no miembro y calculará el precio total correspondiente.
- Recopilación detalles de reserva: El sistema permitirá introducir los detalles necesarios para completar una reserva:
  - Detalles de huésped.
  - Detalles de huésped principal basados en el campo anterior.
  - Detalles de llegada.
  - Detalles de tarjeta de crédito.
- Validación del formulario: El sistema recogerá las entradas introducidas y las comprobará de acuerdo a las reglas del negocio. Si alguno de los valores no pasa la validación el sistema detendrá el proceso de pago y devolverá un mensaje de error, indicando los campos pertinentes.

#### Requisitos no funcionales:

- Diseño adaptativo: La página debe mantener la misma funcionalidad, adoptando el diseño visual pertinente dependiendo del dispositivo usado para acceder a la misma. La página debe hacer uso del mismo código independientemente de dicho dispositivo.
- Mapas: La página debe presentar los resultados de búsqueda y los detalles de hotel representados en Google Maps [11].
- Gestión de interfaces de formulario nativos: En entornos móviles el sistema operativo debe ser el encargado de gestionar la entrada de datos.
- Gestión de idioma de página: El sistema debe habilitar mecanismos para actualizar toda referencia escrita al idioma de la página.
- Gestión de imágenes: El sistema debe ser capaz de referenciar tamaños máximos de imagen basados en la resolución de la página.
- Compatibilidad entre navegadores: Conforme a los interfaces para sistemas operativos para IE7+, Android 2.2+, iOS 5+ y navegadores modernos (Firefox, Chrome, Safari y Opera).

La experiencia de BoomWorks con la tecnología de *responsive design* hasta el momento se había limitado a proyectos de pequeña envergadura y módulos promocionales de sitios existentes. Asimismo, como parte del proyecto YHA era necesario desarrollar funcionalidad *back end* desligada e independiente del componente de presentación. Por otro lado, y debido a la metodología de diseño iterativo seguida por BoomWorks, era posible obtener parte de los diseños de interfaz durante el ciclo de diseño sin que la estructura *back end* o la totalidad de los requisitos funcionales estuviese presente.

Por lo tanto se decidió abordar la implementación de la página de la siguiente manera: El desarrollo *back end* en EKTRON (.NET) y el desarrollo *front end* se llevarían a cabo de manera independiente. Todos los componentes de parte cliente se desarrollarían como plantillas modulares en un prototipo en PHP y más tarde se integrarían en .NET.

Esta decisión tenía como objetivo:

- Crear un entorno controlado para el desarrollo y testeo. El proyecto YHA exponía BoomWorks a campos poco explorados internamente. El diseño basado en *responsive design* era la piedra angular de la propuesta, cuyo peso caía casi enteramente en la parte del navegador. Tanto el estado de la tecnología *responsive* a través de diferentes navegadores de escritorio y móvil como las soluciones alternativas (*fallbacks* y *polyfills*) en 2012 estaban en proceso de maduración. El prototipo *front end* proveía un entorno de desarrollo con variables bajo control y eliminaba elementos de incertidumbre externos.
- Aprovechar al máximo la sinergia con el diseño iterativo. Secciones de la página podían empezar a ser desarrolladas sin esperar al conjunto de diseños ni al estado del *back end*. De la misma manera, las páginas completadas podían ser

utilizadas en testeo de guerrilla para ser validadas por usuarios a medida que se iban completando.

- Independencia de las restricciones del CMS. En proyectos al uso el sistema de gestor de contenidos (CMS) tiene una serie de mecanismos predefinidos para publicar y actualizar contenido que afectan a la estructura de la página. En un sistema adaptativo la estructura de la página es fundamental para la correcta visualización de diseños en diferentes resoluciones. Al desarrollar de los componentes *back end* para adaptarse a las plantillas construidas y validadas nos aseguramos de que la estructura del sistema no se vea comprometida.
- Mejor comunicación con el desarrollo *back end*. Al apartarse de la práctica tradicional de realizar el desarrollo de manera conjunta sobre el mismo conjunto de ficheros, se adoptó una estrategia de desarrollo modular basada en el formato de consumición óptima por parte del sistema EKTRON.

## 2. Diseño de la arquitectura

Como se ha visto el procedimiento a seguir fue el de crear un prototipo en PHP para validar los elementos front end. Dicho prototipo no puede ser construido de manera arbitraria ya que ha de ser transportado al entorno EKTRON.

El formato sugerido por los desarrolladores de *back end* fue el de componentes modulares, fácilmente convertibles en *widgets*.

### 2.1 Páginas y módulos del prototipo

El prototipo se compone de las siguientes páginas.

Secciones de búsqueda de hoteles:

- Homepage.php: Página de inicio. Contiene mapa de búsqueda por Estados (geolocalización) y carrusel con hoteles más populares. No contiene calendario.
- SearchResults.php: Página de resultados de búsqueda por geolocalización. Contiene mapa con *markers* de resultados.
- TextSearchResults.php: Página de resultados de búsqueda por entrada de texto.
- HostelDetails.php: Página de detalles del hotel. Contiene mapa con localización, reseñas de viajeros, detalles del hotel y carrusel con fotos de las instalaciones.

Secciones de reserva de hotel (no contienen calendario):

- SelectRoom.php: Contiene los tipos de habitación disponibles para las fechas y el hotel elegidos.
- RoomSelected.php: Permite introducir el número de huéspedes y su estado de afiliación con YHA. Calcula el precio total a pagar.
- PaymentDetails.php: Recoge los datos de los huéspedes y de pago.

- Confirmation.php: Página final del proceso. Informa al usuario de que la reserva se ha realizado con éxito, presenta datos de reserva, detalles del hotel y de contacto y mapa con la localización del hotel.

Los diferentes componentes de las páginas se pueden diferenciar en tres tipos de módulos.

Módulos de estructura:

- Header.php: Contiene la cabecera HTML y la cabecera de estructura de página.
- Footer.php: Contiene el pie de estructura de página, los scripts de JavaScript de página y los componentes no asociados a elementos (componentes dinámicos, pop-ups).
- Módulos de formulario de reserva:
  - ArrivalDetails.php
  - GuestDetails.php
  - MainContact.php
  - PaymentDetails.php
  - PaymentDetailsErrorOverview.php
  - SelectGuests.php
- ProgressBar.php: Contiene los pasos del proceso de reserva.

Módulos funcionales:

- Map.php: Gestiona la presentación y funcionalidad de mapas Google Maps.
- Calendar.php: Gestiona la presentación y funcionalidad de selección de fechas.
- SearchByText.php: Gestiona la presentación y funcionalidad de la búsqueda de hoteles.

Módulos de contenido:

- HostelDetails.php: Contiene los detalles del hotel.
- HostelOverview.php: Contiene los detalles principales del hotel. Nombre, precio, lugar.
- ImageRotator.php: Contiene las imágenes del carrusel.
- AvailabilityRoomList.php: Contiene las habitaciones disponibles para unas determinadas fechas y hotel.
- BookingDetails.php: Contiene los detalles de la reserva.

Los módulos funcionales son acompañados por el script JavaScript que gobierna su comportamiento. En el caso de los otros módulos, si hay alguna interacción o funcionalidad menor gobernada por JavaScript ésta se incluye a nivel de página.

En ambos casos el script PHP `<?php scripts(); ?>`, ejecutado al final de la página recoge las referencias a los scripts y los ejecuta en última instancia para optimizar la carga de la web [12].

Así mismo, y en base a la funcionalidad en cuestión, el módulo puede necesitar parámetros para definir las particularidades de la situación y para adaptar elementos del lenguaje al idioma de la versión del sitio. Dichos parámetros se pasan como atributos del objeto global “YHAMobile” declarados localmente en el cuerpo de la página:

```
<?php include 'modules/Map.php'; ?>

<script>
// Map data
// User Control codebehind will generate this
YHAMobile.mapData = [{
    latlng : {
        lat: -32.9291279,
        lng: 151.7851346
    },
    infoWindow : '<p id="infoWindowContent"><a href="">Newcastle
Beach</a><br>30 Pacific Street<br>Newcastle , 2300<br>( +612) 4925
3544</p>',
    id : 4294968358
}];
</script>
```

*Código 1 Ejemplo de paso de parámetros a la funcionalidad de módulo*

## 2.2 Patrón de diseño Módulo en JavaScript

El patrón de diseño Módulo [13] es un mecanismo conocido y extendido de encapsular funcionalidad de manera robusta, privada y modular en aplicaciones JavaScript. Fue creado por Richard Cornford en 2003 y popularizado por Douglas Cockford.

El patrón Módulo se utiliza para simular funcionalidad de clase en el lenguaje JavaScript, que carece de la estructura de tipado fuerte de otros lenguajes. De esta manera JavaScript permite crear un objeto al que se le asignan variables y funciones de cariz privado y público.

Las razones para adoptar dicho patrón son las siguientes:

- Protección mediante encapsulado: El patrón módulo nos permite encapsular la funcionalidad en su entorno de ejecución particular, protegiendo sus variables y funciones internas del resto de scripts presentes en la página. Esto se traduce en módulos independientes y que no entran en conflicto.
- Interfaz público: El control UpdatePanel [14] de .NET elimina los eventos JavaScript de sus elementos DOM asociados [15][16]. El patrón módulo permite exponer al entorno global variables y funciones al extender el objeto YHAMobile de tal manera que dichos eventos se pueden volver a asignar a la estructura web.

- Importación: El objeto YHAMobile que actúa como patrón es declarado previamente en el cuerpo HTML. Dicho objeto se puede extender con atributos, que a su vez serán accesibles desde el interior del patrón, ya que la función de creación pasa el propio objeto YHAMobile como parámetro.
- Exportación y extensión: El patrón no sólo recoge el objeto YHAMobile declarado sino que le atribuye toda la funcionalidad expresada en su interior, y lo asigna a una variable global de su mismo nombre. En términos prácticos el objeto se ha actualizado con una serie de funciones y atributos, lo que permite un desarrollo modular de la funcionalidad, ya que el objeto no se destruye o sobrescribe sino que se extiende.

El código que implementa el patrón descrito es el siguiente:

```
var YHAMobile = (function(_this, $, window){
    _this.initialise = function(){
        $(function(){
            _this.publicFunction();
        });
        return _this;
    };

    _this.publicFunction = function(){
        privateFunction();
    };

    var privateFunction = function(){

    };

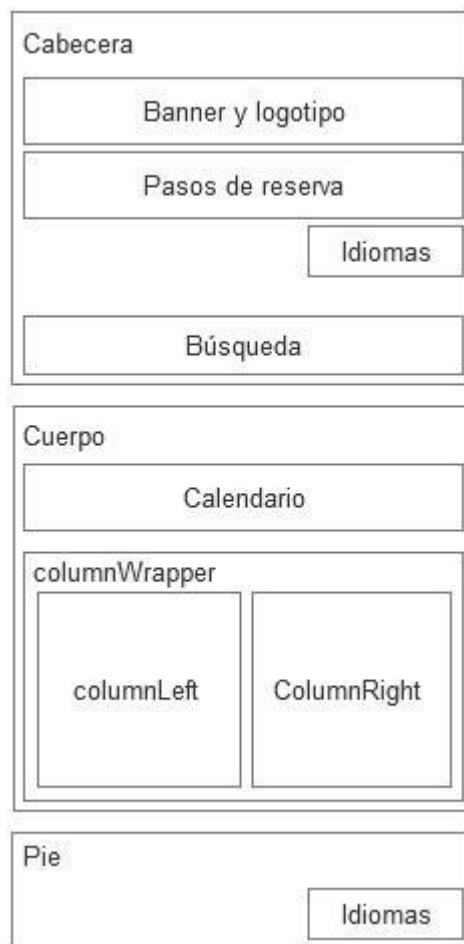
    return _this.initialise();
})(YHAMobile || {}, jQuery, window);
```

*Código 2 Estructura de la implementación del patrón de diseño módulo*

## 2.3 Estructura de la arquitectura de página

La página sigue una estructura clásica de cabecera, cuerpo y pie.

La filosofía seguida a la hora de presentar el contenido es utilizar una columna centrada con dos subcolumnas donde se reparte contenido y funcionalidad en la resolución de escritorio.



*Figura 1 Esquema de la estructura de la página*

Para la versión móvil dichas subcolumnas colapsan en un único componente central y la columna principal pierde sus propiedades como elemento fijador de anchura y adopta un comportamiento fluido.

El patrón visual para los diferentes partes de la estructura de la página es mantener el estilo a lo largo de la anchura de la página. Para conseguir este efecto, los elementos correspondientes han sido envueltos en contenedores que abarcan el 100% de la extensión de la página.

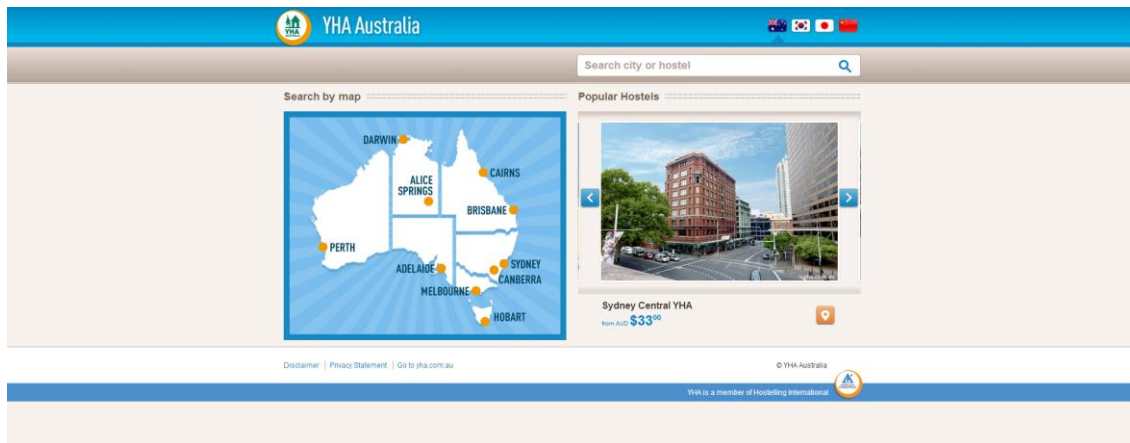


Figura 2 Ejemplo de estilos extendidos a través de la anchura de la página

La cabecera contiene dos conjuntos de elementos diferentes: El banner y logotipo de la página, y los pasos del proceso de reserva.

También contiene la selección de idioma en la resolución de escritorio. Dicha selección desaparece entornos móviles y se transfiere al pie. Ya que esta recolocación de la funcionalidad es drástica en términos de posicionamiento dentro de la estructura de la página, la solución seguida es duplicar el elemento, dando la visibilidad a uno u otro elemento.

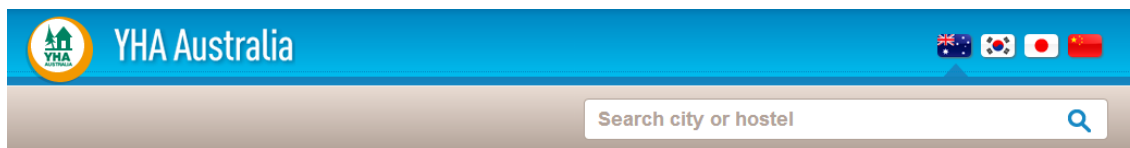


Figura 3 Cabecera de escritorio

El último elemento que contiene la cabecera es el del campo de búsqueda. El posicionamiento de este elemento en resoluciones de escritorio parece pertenecer al cuerpo principal de la página, junto con la funcionalidad de calendario. Sin embargo, su diferente naturaleza se hace patente en la versión móvil, donde aparece debajo del título de la página.



Figura 4 Cabecera móvil

El cuerpo de la página contiene la sección de calendario, que no está incluido en la columna principal al poseer características de estilo diferentes al resto de elementos.

A continuación aparece el elemento contenedor *columnWrapper*, que fija el cuerpo en una resolución establecida para entornos de escritorio.

En su interior los elementos se ordenan en las dos subcolumnas previamente mencionadas por medio de la propiedad CSS *float* [17]. Las clases *columnLeft* y *ColumnRight* establecen qué valor de la propiedad se aplica.



El pie de la página cierra la estructura, con un contenedor que aplica los estilos a lo ancho de la resolución. Como se ha mencionado anteriormente, el pie posee una versión de la funcionalidad para elegir idiomas, que aparece en entornos móviles en lugar de su equivalente en la cabecera.

La página a su vez está envuelta en un elemento HTML *form*, para simular y adaptarse a las restricciones del entorno .NET en el que será construida. Dicha configuración es relevante ya que el lenguaje HTML no permite anidar elementos *form* [18], presentes en los pasos de reserva.

## 2.4 Gestión de resoluciones de dispositivo

La propuesta principal de este proyecto es aunar bajo una misma base de código contenido y funcionalidad enfocado a dispositivos diferentes.

El proyecto YHA tiene dos diseños de interfaz:

- Móvil: Diseñada para una anchura de 320px basada en el iPhone, con estructura fluida. Los elementos de la página se ensanchan para abarcar el ancho del dispositivo contenedor.
- Escritorio: Diseñada entorno a una columna central de 1024px de anchura. Estructura fija. Los elementos internos adoptan anchuras establecidas. Las resoluciones superiores a 1024px centran la columna y dejan espacio a los lados.

Entre estos dos diseños se debía cubrir todo el espectro de resoluciones posibles. Dado que el diseño móvil está basado en una estructura fluida, se decidió llevar el umbral de las hojas de estilo a la resolución de escritorio y no al revés:

```
/*      Desktop      */  
@media only screen and (min-width: 1024px) { ... }
```

*Código 3 Ejemplo de break point para resolución de 1024px especificado con media queries*



# Implementación

A la hora de abordar la implementación del prototipo, se siguió el siguiente orden:

- Diseño móvil. Se decidió empezar por el formato móvil al ser el diseño con más incertidumbres.
- Adaptación a diseño de escritorio. A continuación se añadieron los estilos y elementos DOM de soporte propios de la resolución de escritorio para navegadores modernos.
- *Graceful degradation*: Se pasó a implementar arreglos y *fallbacks* para aquellos navegadores con menor soporte de prestaciones.

## 1. Campo de búsqueda

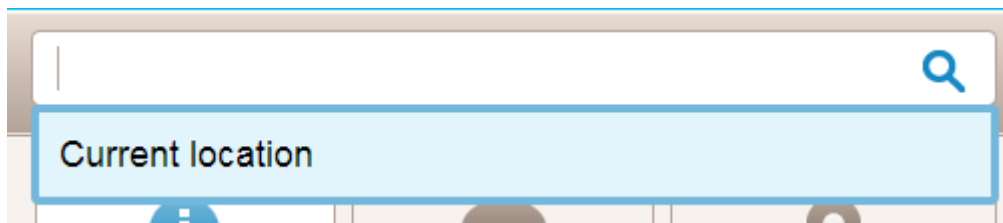


Figura 5 Campo de búsqueda

Uno de los requisitos principales era el de proveer a los usuarios de una herramienta inteligente de búsqueda de hoteles.

Para lo cual se diseñó e implementó un campo de búsqueda único con triple funcionalidad. El elemento proporciona búsquedas por palabras clave, palabras autocompletadas y localización GPS.

A nivel de estructura se usó un elemento de formulario *input* de tipo *text*.

A la hora de abordar la funcionalidad se utilizó el plugin de terceros *jquery.autocomplete.js* [19], parte de la librería *jquery-ui*.

El plugin se encarga de generar un conjunto de sugerencias basada en una lista de pares etiqueta-id. El código fuente del plugin fue modificado para expandir dicha lista a cualquier número de campos. La utilización que se le dio fue incluir un campo *url* con la dirección sobre la que ejecutar la búsqueda.

Dado que el plugin también ejecuta un mapeo de los eventos de interfaz y para no introducir competencia entre manejadores de eventos (*handlers*) se decidió utilizar los eventos nativos del plugin como base.

El módulo JavaScript recibe como parámetros la lista de valores a autocompletar y plantillas de las diferentes direcciones de búsqueda, ya que son gestionadas por separado en el sistema *back end*. También recibe la cadena de texto inicial para habilitar la traducción:

- `autocompleteData`: vector con pares de etiqueta, *url*. La *url* indica la dirección a ejecutar si se buscan hoteles en base a la selección de la etiqueta.
- `textSearchURL`: dirección a ejecutar la búsqueda utilizando texto libre.
- `locationSearchURL`: dirección a ejecutar la búsqueda por geolocalización.
- `currentLocationText`: texto del elemento de búsqueda para traducción.

El comportamiento del campo de búsqueda es el siguiente:

- Si el usuario interactúa con el campo por primera vez se muestra la opción de búsqueda por geolocalización. Esta opción accede a la API de geolocalización de HTML5 [20] y ejecuta búsqueda por posición con los valores de longitud y latitud correspondientes.
- Si el usuario escribe más de tres caracteres, y mientras haya coincidencia, el plugin *autocomplete* entra en efecto, sugiriendo una lista de términos. Al seleccionar una de las sugerencias se ejecuta la búsqueda basada en el valor *url* asociado al término elegido.
- Si la entrada de datos del usuario no encuentra coincidencias y se ejecuta la búsqueda, la funcionalidad ejecutará dicha búsqueda en la dirección de palabras clave especificada en `YHAMobile.textSearchURL`.

## 2. Calendario



Figura 6 Calendario de selección de fechas

La página incluye la posibilidad de filtrar resultados por fechas o seleccionar las fechas de un hotel en particular. La selección de fechas se lleva a cabo a través de un

calendario posicionado dentro de la página que ofrece la posibilidad de seleccionar la fecha de entrada y la fecha de salida.

Internamente se estimó que una funcionalidad de calendario dinámica y adaptada a estructuras *responsive* podía ser reutilizada en otros proyectos. Por lo tanto se decidió implementar dicha funcionalidad de manera agnóstica respecto al proyecto YHA. Para aumentar la portabilidad y la encapsulación se implementó en formato de plugin jQuery. Con esta filosofía se puede asignar la funcionalidad de calendario a cualquier elemento genérico de la página. El mecanismo seguido en otros módulos asigna la funcionalidad pertinente a elementos conocidos y especificados con literales en el cuerpo del módulo.

El módulo que gestiona el calendario por tanto invoca dicho plugin y lo suplementa con los parámetros necesarios.

Inicialmente la funcionalidad de calendario iba a ser más extensa y potente en lo referido a la gestión de la disponibilidad. El plugin está diseñado para incluir información sobre diferentes estados de disponibilidad: completo, limitado, llamar para comprobar. De la misma manera, la fecha de salida no estaba limitada. Restricciones técnicas por parte de YHA eliminaron dichos estados y limitaron la fecha de salida a los siguientes 20 días.

El plugin toma los siguientes parámetros:

- `checkIn`: fecha de entrada.
- `checkOut`: fecha de salida.
- `today`: fecha actual.
- `availability`: vector de disponibilidad de fechas. Funcionalidad no usada en el proyecto.
- `loadingUrl`: imagen a enseñar mientras se ejecuta la función de respuesta.
- `clearDatesEnabled`: *flag* de la funcionalidad de resetear el calendario. No usada en el proyecto.
- `clearDates(response)`: función de respuesta de la operación de resetear el calendario.
- `callback(response)`: función de respuesta del plugin.
- `language`: objeto con cadenas de texto para traducir el contenido del plugin.

El plugin presenta la siguiente funcionalidad:

- Se presenta un calendario al usuario, centrado en la fecha actual. El usuario puede avanzar en el tiempo navegando por meses.
- Al seleccionar una determinada fecha se limita la interacción y se modifica la presencia visual de los días no pertenecientes a los 20 siguientes. Las casillas de la cabecera reflejan la selección.
- Al seleccionar la siguiente fecha el sistema actualiza las casillas y ejecuta la selección.

- Si el usuario accede al calendario de nuevo, las casillas mantienen la información sobre la reserva y el usuario puede seleccionar nuevas fechas o eliminar el filtro.

El plugin hace uso de *moment.js* [21], una librería de gestión de fechas que abstrae el formato nativo JavaScript y le dota de funciones aritméticas.

El valor de respuesta del plugin está parametrizado para garantizar su independencia. El plugin acepta una función *callback* que será ejecutada al finalizar la interacción. Dicha función posee un parámetro *response* con los valores de la selección y el número de noches.

### 3. Gestión de mapas

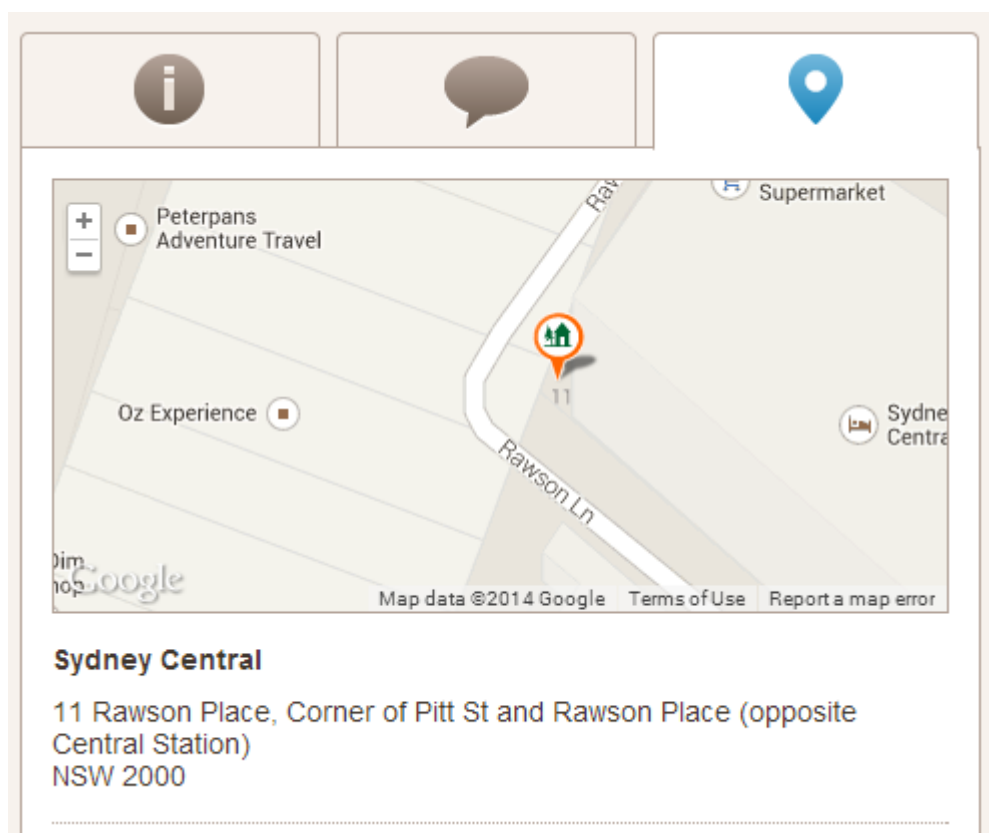


Figura 7 Mapa de detalles de hotel

El proyecto incluye la presencia de mapas para representar gráficamente el número y localización de resultados de búsqueda y para dar información de localización sobre un hotel seleccionado.

El módulo toma los siguientes parámetros:

- MapData: Vector con pares de valores de resultados a colocar en el mapa:
  - latlng: coordenadas del hotel.
  - infoWindow: código HTML y contenido de la ventana a desplegar si se interactúa con su posición el mapa.
- mapMarkerURL: dirección de la imagen para los *markers*.
- enableMapText: texto presente en la capa de protección móvil para su traducción.

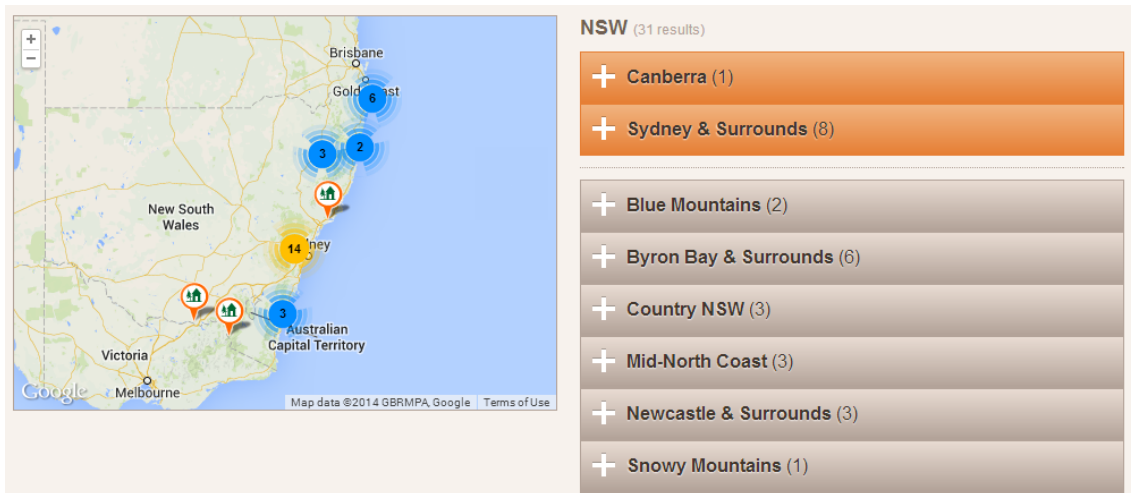


Figura 8 Mapa de resultados de búsqueda

El módulo de mapas es una abstracción y configuración de la API de Google Maps:

- Inicializa la librería de Google Maps y configura el elemento mapa con variables internas con valores de gestión general. La implementación ofrece la posibilidad de extender dichos valores vía parámetros para adaptarse a las necesidades de la página.
- Itera sobre la lista de marcadores pasada como parámetro, les asigna una referencia interna y los posiciona en el mapa.
- Gestiona agrupaciones de marcadores según los niveles de zoom del mapa y la concentración de marcadores en un área.
- Presenta una capa de protección en entornos móviles para evitar que el mapa capture los eventos de interacción con la pantalla, dificultando la navegación. Dicha capa desaparece cuando el usuario hace tap para utilizar el mapa.

La funcionalidad hace uso del plugin *markerclusterer.js* [22] para gestionar la agrupación de marcadores.

#### 4. Validación de formularios de la fase de reserva

Para la validación del formulario y la habilitación de atributos de HTML5 de formulario no soportados se utilizó el plugin de terceros h5validate [23] de Eric Elliot. Dicho plugin analiza los atributos de los elementos de formulario, les asigna parámetros de evaluación donde corresponda y captura el evento de envío del formulario para evaluar su validez.

The image shows a web form for booking confirmation. It is divided into two main sections: 'Arrival details' and 'Payment details'. The 'Arrival details' section includes a dropdown menu for 'Arrival time' and a text area for 'Comments (optional)'. The 'Payment details' section includes a dropdown menu for 'Payment method', three text input fields for 'Card number', 'Name on card', and 'Security code', and a checkbox for 'Yes, I have read the Booking Terms and Conditions'. A green button labeled 'Confirm payment' is at the bottom.

**Arrival details**

✓ Arrival time

Comments (optional)

**Payment details**

✓ Payment method

✓ Card number

✓ Name on card

Security code

✓ mm/yy

☐ Yes, I have read the [Booking Terms and Conditions](#) for the hostel I am booking, and I am aware that a cancellation more than 24 hours in advance will incur a 5% cancellation fee, and a cancellation of less than 24 hours in advance will incur a cancellation fee equivalent to one overnight stay per person.

**Confirm payment**

Figura 9 Detalle del formulario de pago

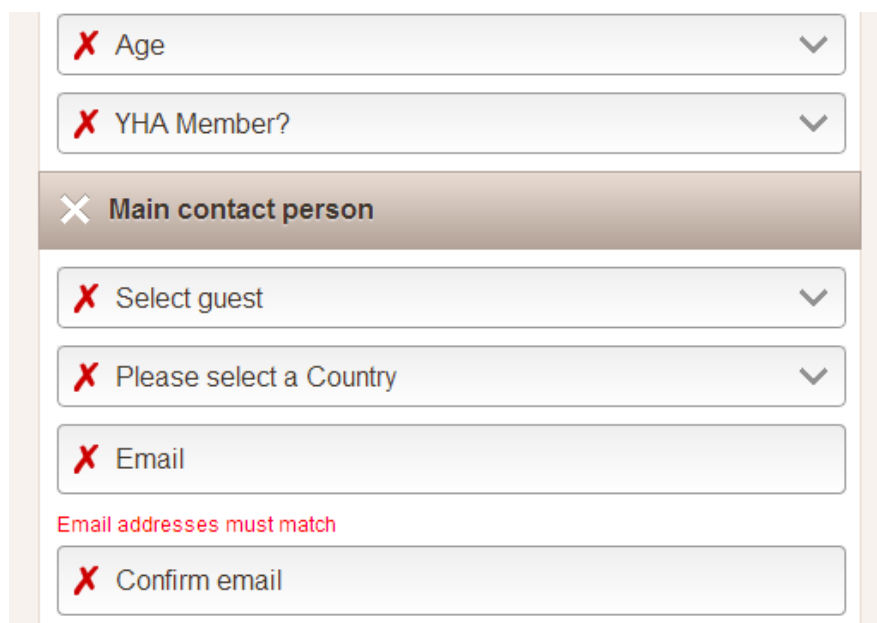
En este caso el atributo HTML5 utilizado por defecto fue el de *required*, que indica que el campo debe contener información.

Otros elementos requerían de condiciones de evaluación más robustas o situacionales. Cuando dicha condición podía ser expresada o recogida por una expresión regular, se utilizó el atributo HTML5 *pattern* al que el plugin daba soporte.



Condiciones más complejas, como la validez de la tarjeta de crédito, el género del huésped en relación al tipo de dormitorio elegido o la coincidencia de emails introducidos se implementaron específicamente, haciendo uso de la funcionalidad de inyección de funciones de validación personalizadas del plugin:

- cardNumber: valida el número de la tarjeta de crédito.
- securityCode: valida el código de seguridad de la tarjeta de crédito.
- expiryDate: comprueba la fecha de expiración.
- matches-prev: comprueba que el campo es igual al campo similar anterior. Utilizado para validación de direcciones de correo electrónico.
- room-age: comprueba que menores de 17 años no estén en habitaciones mixtas,
- room-gender: comprueba que los huéspedes no estén en habitaciones de género único distinto al suyo.



The image shows a vertical stack of form fields. Each field has a red 'X' icon in the top left corner, indicating a validation error. The fields are: 'Age' (a dropdown menu), 'YHA Member?' (a dropdown menu), 'Main contact person' (a section header with a red 'X' icon), 'Select guest' (a dropdown menu), 'Please select a Country' (a dropdown menu), 'Email' (a text input field), and 'Confirm email' (a text input field). Below the 'Email' field, there is a red text message: 'Email addresses must match'.

*Figura 10 Ejemplo de validación de formulario no superada*

La funcionalidad del módulo toma los siguientes parámetros:

- validationFormID: Indica que elemento es el responsable de recoger los valores del formulario.
- roomType: en la fase final de la reserva especifica qué tipo de habitación se ha elegido, seleccionada en el paso anterior.

Se modificó el plugin h5validate para incluir la funcionalidad de asignar un elemento padre para el formulario, indicado con el valor de validationFormID, ya que las páginas .NET implementan el elemento *form* a nivel de página.

## 5. Graceful degradation

Dado que la gran mayoría de los visitantes asiáticos --en especial China, que provee gran parte del segmento de mercado-- utilizan versiones antiguas de Internet Explorer se decidió desarrollar el proyecto con el soporte de IE7 y versiones superiores en mente.

A través de la metodología conocida como *Graceful degradation* [24] se habilita un sistema para funcionar en otro con menos soporte de características, entendiendo que se llega a un compromiso y una serie de mínimos aceptables.

Sin embargo, dichas versiones presentan una serie de problemas:

- *Media queries*: Las *media queries*, pieza fundamental de un sistema *responsive*, como hemos visto, no se introducen en Internet Explorer hasta su versión 9. Para subsanar el problema, la funcionalidad se simuló con herramientas de *front end* a través de la librería JavaScript *css3-mediaqueries.js* [25].
- HTML 5: La mayoría de elementos semánticos HTML5 soportados por IE fueron introducidos también en su versión 9. Para hacer que dichos elementos fueran reconocidos por estos navegadores se utilizó la librería *html5shiv* [26]. Como medida adicional, se dotó de estilo de bloque a los nuevos elementos semánticos en el CSS y se reiniciaron sus estilos [27].
- Gradientes y opacidad: Las versiones 8 y 7 de Internet Explorer no aceptan estilos CSS para gestionar las propiedades de CSS3 relativas a gradientes en las imágenes de fondo ni opacidad de elemento. Para replicar dicha funcionalidad se ha de hacer uso de filtros propietarios que invalidan el CSS estándar:

```
.calendarFilter-overlay {  
    filter: alpha(opacity=50);  
    -ms-  
    filter: "progid:DXImageTransform.Microsoft.Alpha(Opacity=50)";  
}
```

*Código 4 Ejemplo de filtro propietario de Microsoft para implementar transparencia en Internet Explorer*

- Block-inline: Block-inline es un valor de la propiedad CSS *display* [28] que permite utilizar elementos de tipo bloque haciendo que se comporten y se alineen siguiendo las reglas de elementos de tipo línea. Esta propiedad tiene gran utilidad a la hora de posicionar elementos con objetivos estructurales y visuales. Sin embargo un conocido bug de IE7 impide ser presentado apropiadamente. Se resuelve con el siguiente *hack*:

```
{display: inline-block; *display: inline!important; zoom: 1;}
```

*Código 5 Inline-block hack para dotar comportamiento inline-block a elementos en Internet Explorer 7*

Las soluciones se incluyen en la página por medio de etiquetas HTML condicionales para separar los elementos no estándar del código general y preservar su integridad:

```
<!--[if lt IE 9]><script src="js/html5-shiv.js"></script><![endif]-->
<!--[if lte IE 8]><link rel="stylesheet" media="screen,projection"
href="css/style_ie8.css"><![endif]-->
<!--[if lte IE 7]><link rel="stylesheet" media="screen,projection"
href="css/style_ie7.css"><![endif]-->
```

*Código 6 Ejemplo de hojas de estilo cargadas condicionalmente basadas en versiones de Internet Explorer*

## 6. Gestión de imágenes

En la implementación tradicional de *responsive design* se asigna propiedades de estructura fluida a las imágenes, que se encogen y ensanchan manteniendo su relación de aspecto. Sin embargo esto significa que las imágenes tienen que ser transferidas a la mayor resolución a la que puedan ser vistas para que no haya pérdida de calidad.

En términos generales se intentó introducir imágenes con tamaños fijos y reducidos (*i.e.*, miniaturas de resultados), pero las imágenes de los carruseles eran de una calidad y tamaño suficientes como para afectar de manera negativa la descarga y rendimiento de los dispositivos móviles. Por lo tanto se decidió tomar una solución combinada de *front end* y *back end*.

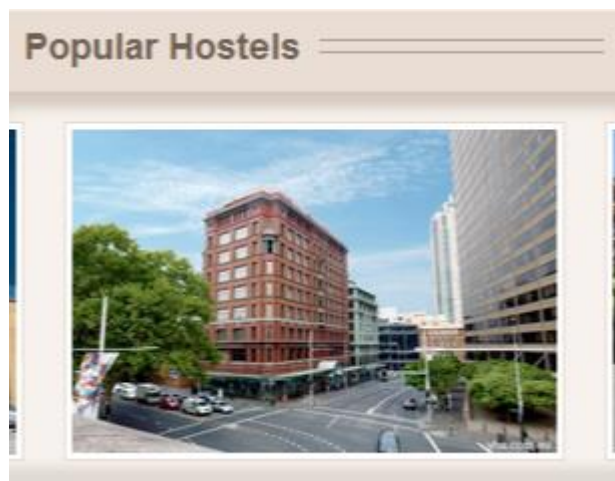


Figura 11 Carrusel de página de inicio redimensionado para entornos móviles

El sistema *back end* implementó una función que reescala una imagen basada en el atributo “query” de su dirección.

El sistema *front end* recoge los elementos de imagen presentes en el carrusel, calcula la anchura de la página para el dispositivo en cuestión y reescribe el atributo *src* añadiendo el parámetro correspondiente como atributo de *query*. De esta manera la imagen descargada será de la resolución apropiada:

```
$('#carrousel img').each(function () {  
    var src = $(this).data('src');  
    $(this).attr('src', src + '?w=' + originalWidth);  
});
```

*Código 7 Demostración de la reescritura dinámica de la referencia a imágenes para adaptarlas a la resolución del dispositivo*

El sistema vuelve a realizar el cálculo si se produce un cambio de orientación en el dispositivo que da lugar a una mayor anchura de página.

## 7. Retos y problemas

La dificultad del proceso de implementación fue fruto de la propia naturaleza de la metodología de diseño adaptativo. Según los requisitos no funcionales el sistema debía asegurar la integridad funcional y visual para IE7+, Android 2.2+, iOS 5+ y navegadores modernos (Firefox, Chrome, Safari y Opera) de sobremesa.

En la práctica este requisito hacía necesario desarrollar para 12 navegadores a través de 5 entornos diferentes, 4 de ellos móviles:

- Internet Explorer 7
- Internet Explorer 8
- Internet Explorer 9
- Firefox
- Chrome
- Safari
- Opera
- Android 2.2.1
- Android 2.3
- Android 4.0.4
- iOS 5
- iOS 6

En este sentido se tuvo que llevar a cabo un exhaustivo ejercicio de disciplina y control. El desarrollo de elementos se realizó de manera progresiva a lo largo de los diferentes navegadores. Una vez finalizada la implementación de un componente se iteraban las implementaciones previas para controlar el impacto de los cambios introducidos. En los casos en los que las diferencias introducidas eran irreconciliables se realizaba una ponderación de los resultados obtenidos por navegador para llegar a una solución de compromiso.

Este trabajo se vio acompañado por la necesidad de realizar sesiones específicas de investigación y estudio en detalle de las particularidades de los motores de presentación y las implementaciones de JavaScript a lo largo de los 12 navegadores, con el objeto de entender la raíz de las diferencias y encontrar la solución menos intrusiva a nivel de sistema para problemas específicos.

Algunas de las soluciones adoptadas se han visto en el apartado anterior y una incidencia especialmente compleja se detalla en el apartado 5.1 de la sección “Pruebas del sistema”.

En líneas generales es esta característica del proyecto la responsable por la complejidad, esfuerzo necesario y horas invertidas en el mismo.



# Pruebas del sistema

En este apartado se presenta el plan de pruebas utilizado para asegurar la calidad del producto final y su conformidad con los requisitos del sistema.

## 1. Estrategia

Las pruebas de sistema se realizaron de manera progresiva a medida que las páginas y sus módulos asociados se iban completando. La validación se realizó a nivel de módulo, ya que varios de ellos aparecen a lo largo del portal y porque esa es la estructura utilizada por los desarrolladores *back end* para consumir las plantillas producidas.

Por cada módulo a comprobar se realizaban pruebas en los diferentes entornos objetivo y sus correspondientes navegadores.

Las pruebas se realizaron de manera manual, ya que las pruebas automatizadas no se acoplaban a la naturaleza dinámica del proceso de diseño y desarrollo.

Al acabar el proceso de creación del prototipo se realizó otra serie de pruebas a nivel de portal para asegurar la consistencia y robustez de la solución.

## 2. Entorno de pruebas

Para facilitar el proceso de prueba se habilitó un sitio web en el servidor de la compañía, BoomShed [29].

Los desarrolladores dispusieron de un entorno de trabajo local donde realizar pruebas y correcciones. Al acabar con un error, el código correspondiente se añadía al repositorio en BoomShed.

El equipo de pruebas tenía a su disposición ordenadores equipados con versiones de los navegadores especificados en los requisitos y varios dispositivos móviles con las versiones correspondientes de iOS y Android.

## 3. Roles

Los perfiles de los participantes en las pruebas del sistema son los siguientes:

- **Desarrollador:** El propio desarrollador del sistema realiza pruebas en su entorno local para comprobar que cumplen los requisitos.
- **Diseñador visual:** El diseñador visual del proyecto comprueba que las páginas producidas se adhieren a los estilos visuales.
- **Diseñador de experiencia de usuario.** El diseñador UX comprueba que los elementos funcionales se comportan de acuerdo al diseño y que el usuario puede realizar las tareas del sistema.
- *Project manager:* El *project manager* vela por la integridad de la solución a lo largo del sistema y comprueba que cumple con los requisitos del cliente.

## 4. Tipos de pruebas

- Pruebas unitarias: Para cada elemento funcional producido se ejecutaron diferentes escenarios de uso para comprobar que la multiplicidad de interacciones posibles no producía errores en el funcionamiento.
- Pruebas de integración: A medida que se completaban las pruebas unitarias se construían las páginas con los módulos funcionales correspondientes y se comprobaba que no se había comprometido la integridad de la página ni las distintas funcionalidades de los módulos.
- Pruebas de sistema:
  - Pruebas funcionales: Una vez completadas las pruebas de integración se recorrió el proceso de reserva de un hotel bajo varios escenarios para comprobar que los diferentes componentes no rompían el flujo de la reserva.
  - Pruebas no funcionales: Se realizaron pruebas de usabilidad, conformidad con el diseño visual, rendimiento de carga de imágenes y de traducción.

## 5. Problemas destacados

### 5.1 Comportamiento errático de la pantalla al interactuar con elementos de tipo input en Android 2.2

Problema:

En móviles Android 2.2 la pantalla saltaba de lado a lado al interactuar con elementos de formulario HTML, haciendo imposible completar el proceso.

Solución:

El problema venía dado por la utilización de la propiedad CSS *backface-visibility* [30]. El rendimiento de una web con animaciones CSS3 mejora considerablemente si se utiliza el motor 3D hardware del dispositivo en lugar del motor nativo del navegador [31]. Para asegurarse de que la animación se realiza por hardware se añade al elemento a animar propiedades que utilicen dicho motor, pero de forma inocua.

Las propiedades a utilizar por sus efectos a través de los diferentes navegadores son *backface-visibility: hidden* y *translateZ(0)*.

Sin embargo, en versiones Android 2.2 y anteriores el motor 3D contenía errores que daban lugar al comportamiento errático descrito [32]. En este caso el padre del formulario era un elemento *compactor* que hacía uso de animación al abrirse.

La solución fue utilizar otras propiedades 3D, siendo *translate3d(0,0,0)* la que proporcionó el resultado deseado.



## 5.2 Google Maps imposibilita la navegación de la pantalla en entornos móviles

Problema:

Google Maps captura los eventos en el mapa en base a sus especificaciones de comportamiento. En entornos móviles dicho mapa abarca casi la totalidad de la pantalla y al interactuar con él, se mueve el interior del mapa y no la página. Esto dificulta mucho la navegación hacia partes inferiores de la página.

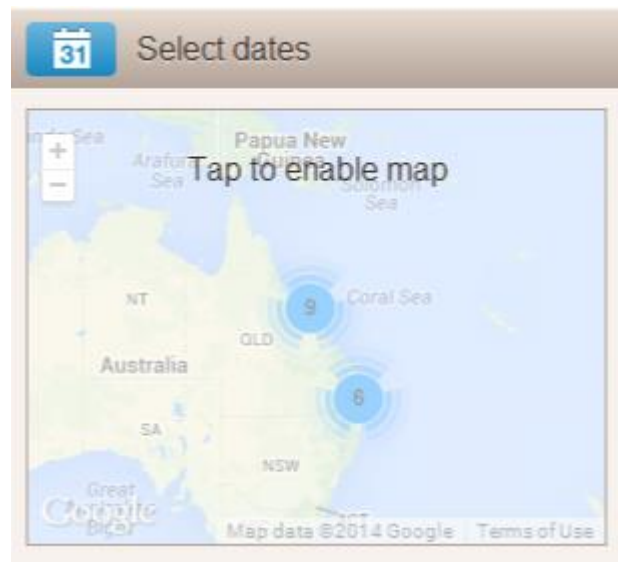


Figura 12 Capa de protección para la interacción con Google Maps

Solución:

Se decidió crear una capa especial para entornos móviles que capturase la interacción con el elemento de manera natural. La capa en sí tiene un mensaje que informa al usuario de que debe hacer *tap* en el elemento si desea utilizar la funcionalidad de Google Maps.

## 5.3 Funcionalidad de restaurado de fechas en el calendario

Problema:

Al probar la funcionalidad del calendario desde distintos escenarios se comprobó que no era posible restaurar el filtrado por fechas.

Solución:

Se incluyó la funcionalidad para restaurar la selección de fechas, se posicionó un link en el calendario para activarla y se añadió el parámetro *clearDates(response)* como función *callback* para facilitar la gestión por parte del sistema *back end*.

## 5.4 Aspect ratio en los carruseles de imágenes

### Problema:

Al cambiar la orientación de los dispositivos móviles los carruseles no se adaptaban a la nueva resolución, rompiendo la estructura y afectando a la navegación.

### Solucion:

El plugin que implementa la funcionalidad de carrusel adoptaba una estrategia de altura fija definida para elemento padre en base a la anchura inicial del componente. La solución fue la de asignar una función al evento de cambio de orientación que recogía la nueva anchura y adaptaba el carrusel a las nuevas dimensiones



# Conclusiones

## 1. Resultados obtenidos

Se puede concluir que los resultados del proyecto son plenamente satisfactorios atendiendo a tres factores:

- Cumplimiento de los requisitos. El sistema cumple con cada uno de los requisitos especificados en la fase de diseño.
- Integración con el sistema *back end*. No hay que olvidar que el objetivo último del sistema es producir plantillas preparadas para ser integradas en el CMS EKTRON. En este sentido la integración con el sistema *back end* se realizó sin problemas y de manera fluida. Las únicas acciones tomadas en el proceso fueron la de convertir ciertas funciones de clase a métodos públicos y la de parametrizar valores de configuración. En ambos casos el cambio se debió a que dichos elementos se gestionan por parte del CMS y debían estar expuestos. La decisión se tomó conjuntamente con los desarrolladores *back end*.
- Comportamiento en producción. El sistema al completo está en pleno funcionamiento y abierto al público a día de hoy cumpliendo con su propósito. La estructura y funcionalidad del sistema permanece intacta, rindiendo al nivel deseado.

En cuanto al proceso y el impacto interno de desarrollar un sistema *front end* con tecnología adaptativa, se puede considerar por las razones expuestas anteriormente que el primer proyecto a gran escala de BoomWorks de estas características ha sido un éxito.

## 2. Lecciones aprendidas

El proyecto ha sido un verdadero ejercicio de aprendizaje en las diferencias y particularidades de los diferentes navegadores y sistemas operativos asociados.

El proyecto me ha proporcionado la posibilidad de estudiar en detalle tecnologías web en sus variantes más actuales, como HTML5 y CSS3. También he abordado el aprendizaje y uso del lenguaje JavaScript como herramienta madura, con estructuras de organización complejas, y no como soporte cosmético o de manipulación del DOM.

El proyecto me ha expuesto a una variedad de particularidades y problemas a través de diferentes navegadores, muchas veces relacionados y contrapuestos. Esta experiencia se ha visto complementada con la necesidad de buscar soluciones a través del estudio de materiales y la investigación de recursos.

Por último, me ha expuesto a la metodología de diseño adaptativo, utilizada ampliamente hoy en día, lo que supone una experiencia de gran valor.

### 3. Líneas de trabajo futuras

En lo que al proyecto YHA se refiere no hay elementos pendientes de mejora o extensión. Los portales *multilingual* y la versión móvil se liberaron en febrero de 2012 y desde entonces el sistema no ha sufrido ningún cambio relevante. La mayoría de las peticiones de cambio por parte del cliente están relacionadas con gestión de contenidos de hoteles y campañas de promoción.

Sin embargo y atendiendo a las necesidades de usuario el sistema se beneficiaría de ciertas mejoras.

Actualmente solo existe un punto de corte entre resoluciones, a los 1024px. En resoluciones intermedias hasta el tamaño móvil la página es funcional pero el diseño visual sufre. Añadir estilos y con sus correspondientes *breakpoints* para tabletas subsanaría el problema.

El portal de YHA tiene como objetivo de negocio posicionarse en mercados extranjeros, en concreto Asia, pero sus servidores están en Australia. El sistema mejoraría sus prestaciones utilizando un CDN [33] para servir el contenido estático desde localizaciones más cercanas al dispositivo que origina la petición.

Como estrategia habilitadora, se pueden estudiar mecanismos para diferenciar de manera orgánica resoluciones a nivel de JavaScript [34]. Dichos mecanismos permitirían introducir funcionalidad específica para resoluciones en particular. A medida que el público se familiariza con web y apps móviles se adopta una serie de modelos mentales y patrones de interacción propios del soporte que no tienen una correlación directa con la web de escritorio. Conocer y controlar los mecanismos anteriormente mencionados posibilitaría cambios en este sentido en un futuro.

A nivel interno y tras esta primera prueba satisfactoria se puede decir que el siguiente paso es abandonar la estrategia de prototipo y pasar a implementar las plantillas en un entorno de desarrollo conjunto utilizando lo aprendido para evitar conflictos.



## Anexos

### Anexo A: Investigación de competidores móviles

Como parte del proceso inicial de análisis y recogida de necesidades y oportunidades para crear los requisitos del proyecto el equipo de diseño de experiencia de usuario realizó un estudio del mercado. Los materiales a estudiar fueron versiones móviles de webs o aplicaciones móviles de reserva de alojamiento.

Hello

### Mobile sites/apps reviewed

1. Airbnb – m.airbnb.com
2. Hostelworld – m.hostelworld.com
3. Kayak – m.kayak.com
4. Hipmunk – App
5. We hostels - App
6. Hotels.com - App

BOOMWORKS.COM.AU

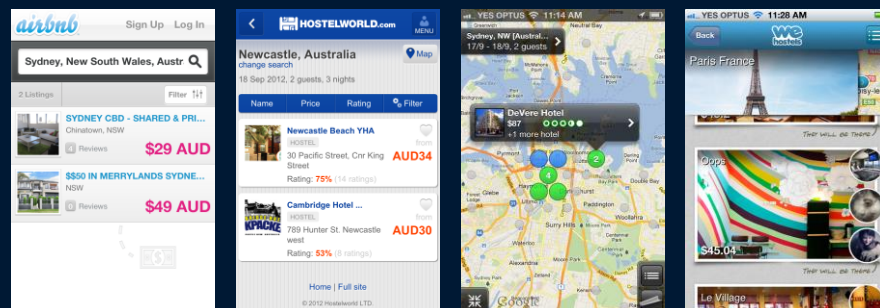
BOOMWORKS 2

## Search hostels and dates



- ✓ Both apps and mobile sites offer search by current location
- ✓ Most of the reviewed sites/apps lets users choose both place and dates before searching. We Hostels and Airbnb allows a search for city and hostel before specifying dates.
- ✓ The Kayak interface has large buttons to add guests and rooms that are easy to interact with.
- ✓ A majority of sites/apps lets the user specify the check in and check out dates while others use arrival date together with number of nights

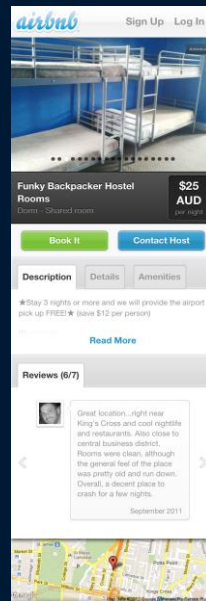
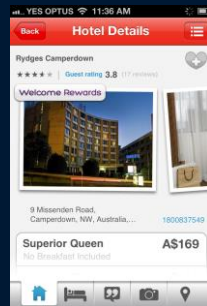
## Search results



- ✓ The most common way to display hotels/hostels is a list with a small images together with some info and price. A couple of apps were a bit more engaging with the use of images, location and social media.
- ✓ Many sites/apps have filters and sorting available across the top. Airbnb have collected all filters in one button that opens up a new page.
- ✓ Both Hostelworld and Airbnb includes an indicator of reviews and ratings.
- ✓ Hostel world lets you "like" a hostel and save it for later.

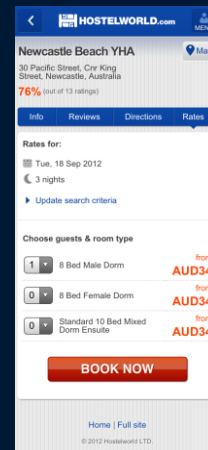
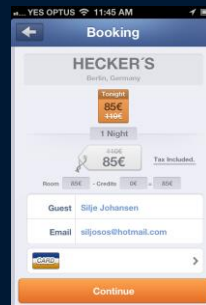


## View hostel details



- ✓ The majority of apps/sites lets you flick through images horizontally in the hotel/hostel details.
- ✓ There is a mix of compactor, tabs and links to separate pages used for information, reviews and location.

## Book rooms

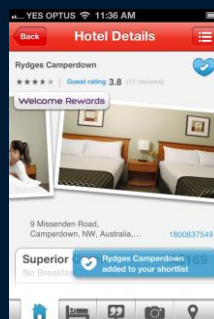


- ✓ The apps/sites for hostels let you navigate the various rooms available before going ahead with the booking.
- ✓ We hostels lets you choose dates at this stage while users have to perform a new search in Hostelworld to change the dates.
- ✓ We hostels lets you choose the number of nights, but also reveals the departure date.

## Book rooms

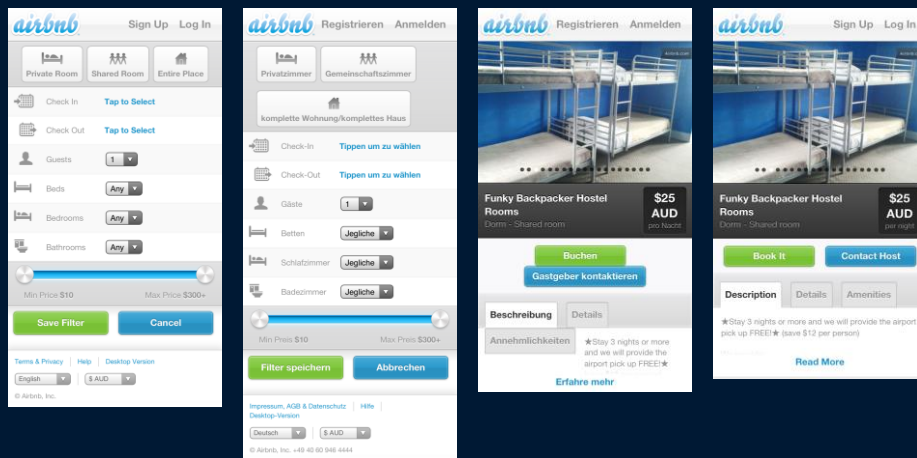
- ✓ Hostel world separates personal details and payment details while we hostels shows everything on one page.
- ✓ Kayak hides information in compactors and users can easily go back and change information if needed.

## Fun and engaging details



- ✓ The Hipmunk mascot does a little dance next to the progress bar while the app loads the search results onto the map
- ✓ We hostels focuses on social media and lets users chat with other users that have booked at the same hostel. They also have an animated menu that pops out when you press the pluss icon.
- ✓ Airbnb uses a large calendar that is easy to use for both check in and check out.
- ✓ You can save hotels to a shortlist by pressing a little heart in the Hotels.com application.

## Multi lingual support



- ✓ The images show how the interface can change drastically depending on language.
- ✓ Adding several tabs or buttons horizontally could present problems especially with languages like German where the words tend to be longer.

## Anexo B: Bocetos de wireframes móviles

El equipo de diseño de experiencia de usuario creó una serie de bocetos para demostrar la estructura y funcionalidad de la página. Estos bocetos se testearon con viajeros en hoteles de la cadena YHA para validar las propuestas.

### B1 Resultados de búsqueda





## B2 Herramienta de filtrado

The sketch depicts a user interface for YHA Australia, specifically a filter tool. At the top, the YHA Australia logo is shown, consisting of a red circle with a white upward-pointing arrow. To the right of the logo, the text "YHA Australia" is written in a bold, sans-serif font. Below the logo and text, there are two links: "BOOK HOSTEL" and "VIEW BOOKINGS". A search bar is located below the links, with a magnifying glass icon on the left and a search icon on the right. The search bar contains the text "Search city or hostel". Below the search bar, there is a "Filter" button with three upward-pointing arrows. To the right of the "Filter" button, there are two icons: a location pin and a list icon. The main filter section contains two date selection boxes labeled "CHECK IN" and "CHECK OUT", each with a "Select" button. Below these boxes is a guest selection control with a minus button, the text "2 Guests", and a plus button. At the bottom of the filter section is a price range slider with a circle on the left labeled "Min price" and a circle on the right labeled "Max price". Below the slider are two buttons: "Save filter" and "Cancel".

YHA Australia

BOOK HOSTEL | VIEW BOOKINGS

Search city or hostel

Filter ↑↑↑

CHECK IN

Select

CHECK OUT

Select

2 Guests

Min price

Max price

Save filter

Cancel

⊕  ⊗

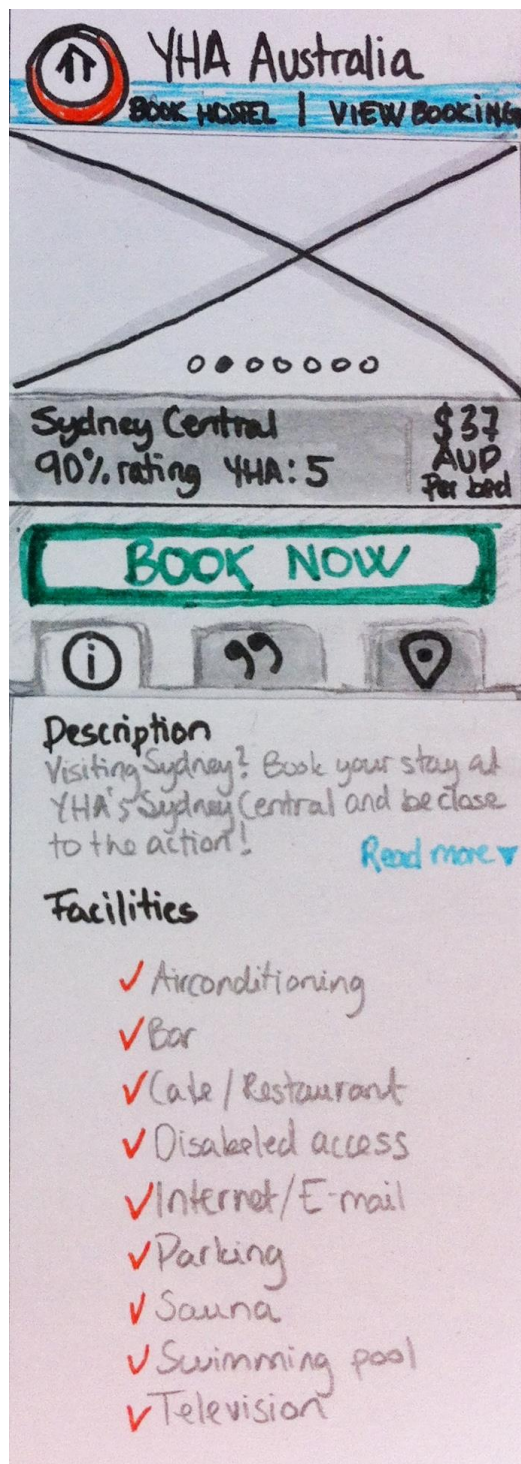
Hostels by state

NSW/ACT  
Northern Territory  
Queensland  
South Australia  
Tasmania  
Victoria  
Western Australia

Q W E R T Y U I O  
A S D F G H J K L ; ' [ ]  
~  
↑ ↓ ← → ↶ ↷  
? 1 2 3 - . @ return



## B4 Detalles de hotel



**CHECK IN** **CHECK OUT**

**17/09/2012**


◀ **September 2012** ▶

Sun	Mon	Tue	Wed	Thur	Fri	Sat
16	17	18	19	20	21	22

limited availability

Not available




**YHA Australia**  
[BOOK HOSTEL](#) | [VIEW BOOKINGS](#)

**Sydney Central YHA**  
 Check in: Mon 17 Sept      Check out: Wed 19 Sept

2 people for 2 nights \$7800 AUD  
Change      AUD ▼

**Personal details**

<input checked="" type="checkbox"/>	JOHN SMITH	
<input type="checkbox"/>	Last name	
<input type="checkbox"/>	Country of residence	
<input type="checkbox"/>	Gender	▼
<input type="checkbox"/>	Age	▼
<input type="checkbox"/>	Arrival time	▼

**Guest 2**  
 Maria Jones Change

**Guest 3**

<input type="checkbox"/>	First name	
<input type="checkbox"/>	Last name	
<input type="checkbox"/>	Gender	
<input type="checkbox"/>	Age	

**Guest 4**

<input type="checkbox"/>	First name	
<input type="checkbox"/>	Last name	
<input type="checkbox"/>	Gender	
<input type="checkbox"/>	Age	

**Payment Details**

<input type="checkbox"/>	Card type	
<input type="checkbox"/>	Card number	
<input type="checkbox"/>	Name on Card	

 Expiration date  

▼

▼

☐ Security code 2/2

By clicking on confirm payment you agree to the Terms and conditions

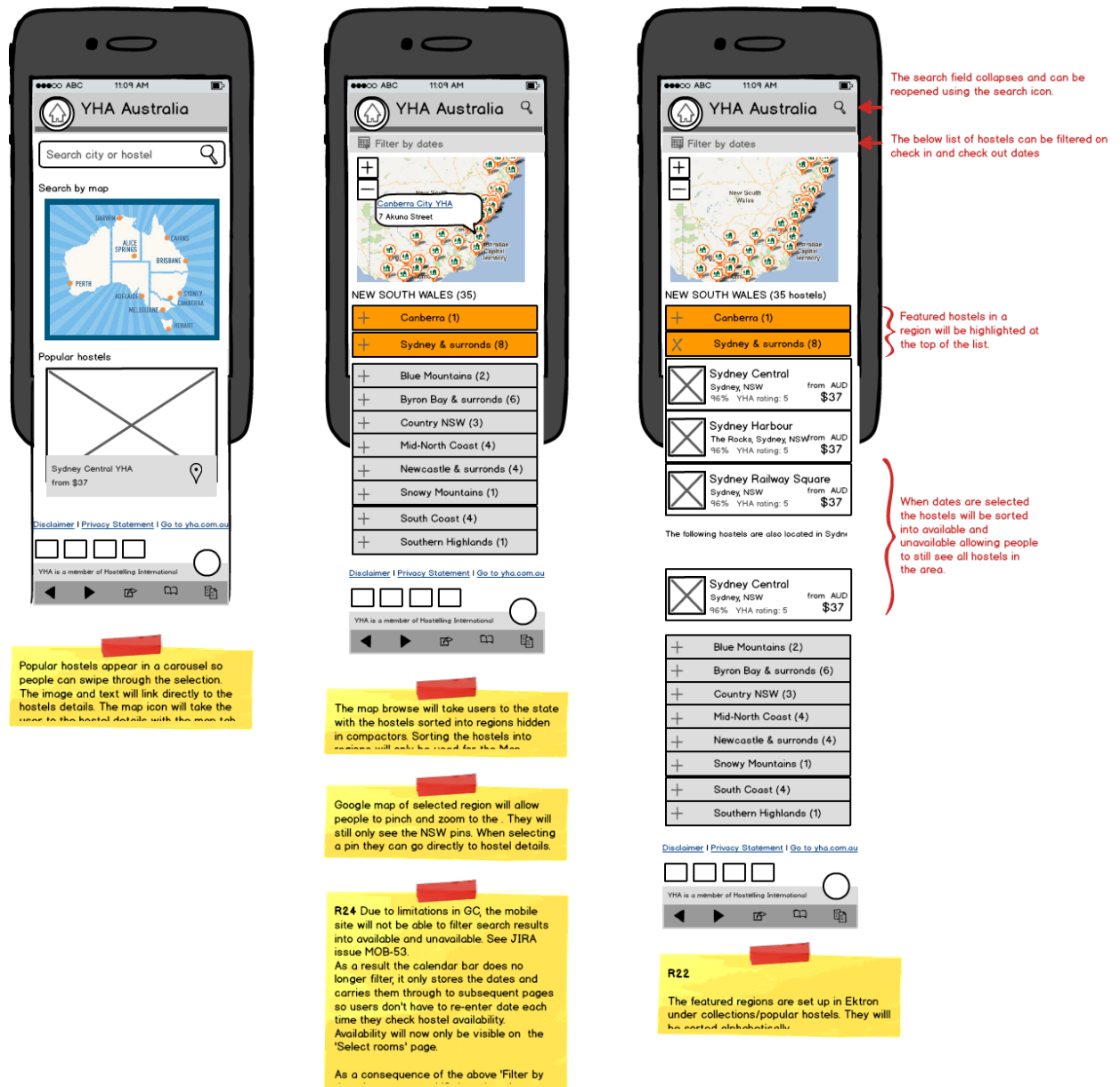
**CONFIRM PAYMENT**

## Anexo C: Wireframes móviles

Como resultado del proceso de diseño se crearon *wireframes* para dispositivos móviles.

### C1 Map browse

YHA mobile wireframes - Map browse - Page 3



## C2 String search

YHA mobile wireframes - String search - Page 4

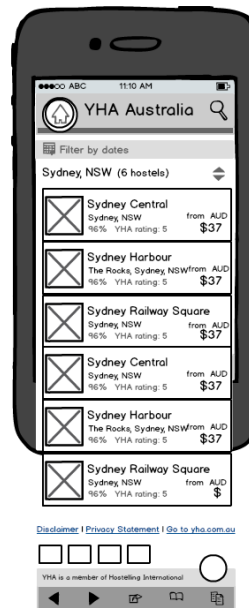


When tapping the search field, users can select current location to view hostels nearby. The search result will be limited to 10 nearby hostels sorted on distance. The results will be displayed the same way as a string search.



A list of suggestions appear as the user starts typing.

**R12**  
Suggestions will show matches in locations above hostels. This example shows suggestions for a location and a hostel. Sydney, New South Wales will return a list of hostels in Sydney while choosing any of the hostels will take the user straight to the hostel details page.



The search results will appear in a list and not sorted into regions like the map browse. Users can sort the search result using the sort button.

**R17**  
Not all hostels may have defined a value for the field 'from price'. In the case where no value is defined we will display 'From AUD \$'.

See example with Sydney Railway Square

**R23**  
Search field hidden when list of results appear. This allows for more results shown above the fold. The search phrase is repeated in the heading.



The sort button reveals the options to sort.

**R1**  
Sort by location is left out for the mobile web. On the main site location sorts results after geographical location to each other. On the mobile this behaviour is easily confused.

**R5**  
Decided to use native selectors on browsers. The native selectors perform better.

**R10**  
Removed the option to sort for YHA grading. This was discussed in an earlier meeting with Ralf.

# C3 Calendar

YHA mobile wireframes - Calendar - Page 5



The calendar icon bar will slide open the calendar. The calendar will act as an overlay and block out the rest of the content until dates are selected. Users can select check in date first, and the check in field will be highlighted to indicate this. Check out and nights will be greyed out until a check in date is selected.



When a check in date is selected the user can either choose a check out date or add the number of nights both fields will be coloured to indicate they are active. The user can only select up to 14 nights from the selected check in date, and nights after this limit will appear greyed out.



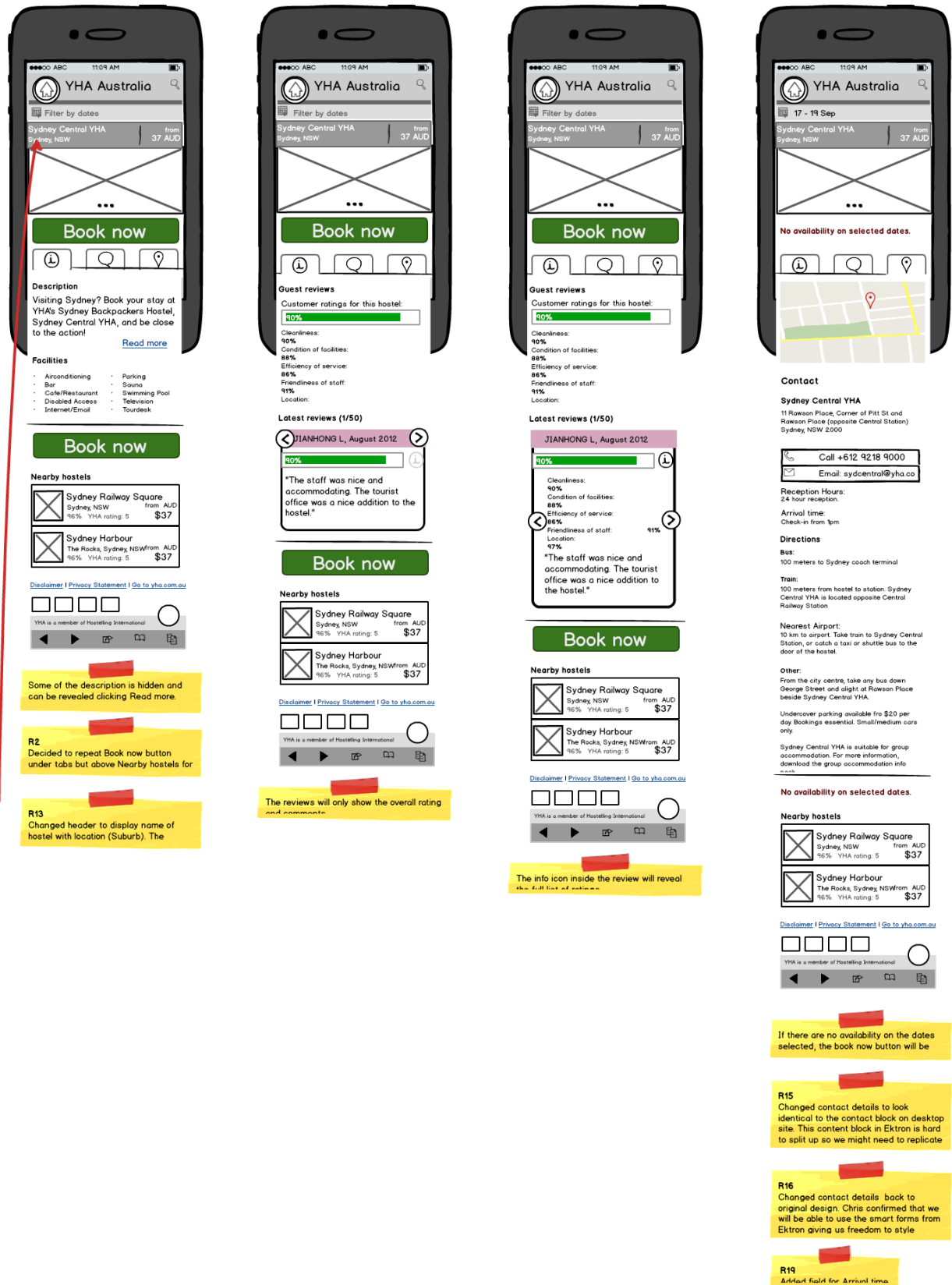
If the user taps the nights section, the native selector will appear



When number of nights or a check out date is selected the calendar will linger for a second before automatically closing. The dates will now show in the bar and users can reopen the calendar by tapping the bar.

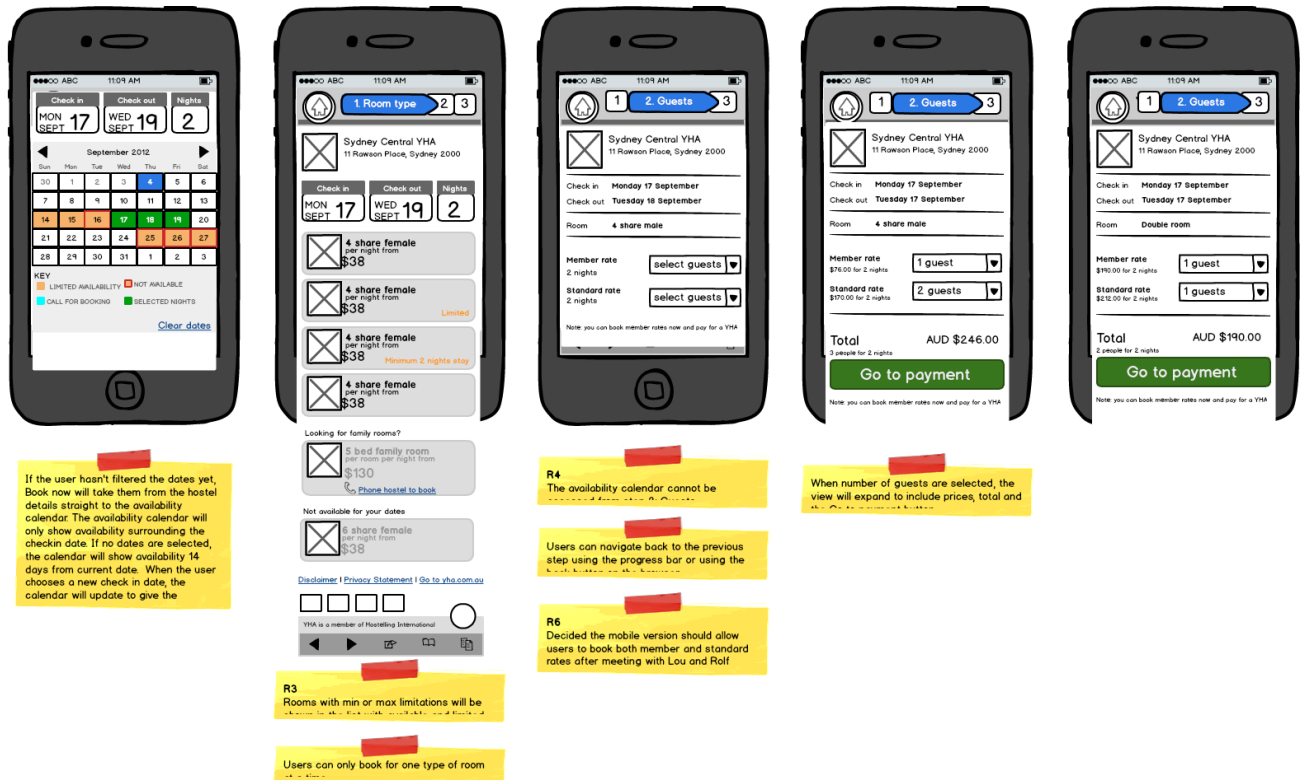
## C4 Hostel details

YHA mobile wireframes - Hostel details - Page 6



## C5 Select rooms and guests

YHA mobile wireframes - Select room and guests - Page 7



## C6 Payment

YHA mobile wireframes - Payment details - Page 8

**Sydney Central YHA**  
11 Rawson Place, Sydney 2000

Check in: Monday 17 September  
Check out: Tuesday 17 September

Room: 4 share male

**Total** AUD \$246.00  
3 people for 2 nights [change](#)

**Guest 1**

First name: [X] Will  
Last name: [X]  
Gender: [X] Female  
Age: [X]  
YHA member: [X]  
Member number: [X]

**Guest 2**  
Aron Smith [edit](#)

**Main contact person**

Select guest: [X]  
Country of residence: [X]  
Email: [X]  
Confirm email: [X]

**Arrival details**

Arrival time: [X]  
Comments (optional): [X]

**Payment details**

Payment method: [X]  
Card number: [X]  
Name on card: [X]  
Expiry date: [X] MM/YY [X]  
Security code: [X]

☐ Yes, I have read the [Booking Terms and Conditions](#) for the hostel I am booking, and I am aware that a cancellation more than 24 hours in advance will incur a 5% cancellation fee, and a cancellation of less than 24 hours in advance a cancellation fee equivalent to one overnight stay per person.

**Confirm Payment**

### Error messages

#### Mandatory fields

All fields with a cross are mandatory and will turn to a green check icon when they are filled out. The confirm button should not appear until all mandatory fields are filled in.

#### Gender/Age

In the case where there are restrictions on the room, this can't be validated before confirm button is pressed. The error message will then appear under the field like illustrated on the left.

- Females cannot stay in Male shares
- Males cannot stay in Female shares
- Under 17's cannot stay in any share rooms

#### Payment issues

- Card details are not correct (Card number, securitycode, name, expiry date etc)
- Insufficient funds
- Problems with payment gateway

Change link takes the user back to previous page (Same as navigating backwards to step 2. on the progress bar.

Will

Gender

You must be male to stay in a 4 share male room

Member number only appears if member is selected from dropdown. Similar to desktop behaviour

Guest 2  
Aron Smith [edit](#)

Once all guest fields are filled out the guest collapses only showing the name.

Arrival time content will be what is defined in the hostel smartform as 'Hostel Reception Opening Times'

Last 3 digits as they appear on the signature side of the card.

Booking Terms and Conditions should appear in a lightbox similar to the one used for Disclaimers. See page 10 - Footer

Confirm payment button should only appear when all fields are filled in

**R8**  
Changed label for choosing credit card type from Card type to Payment method to futureproof the form. PayPal will be added at some point

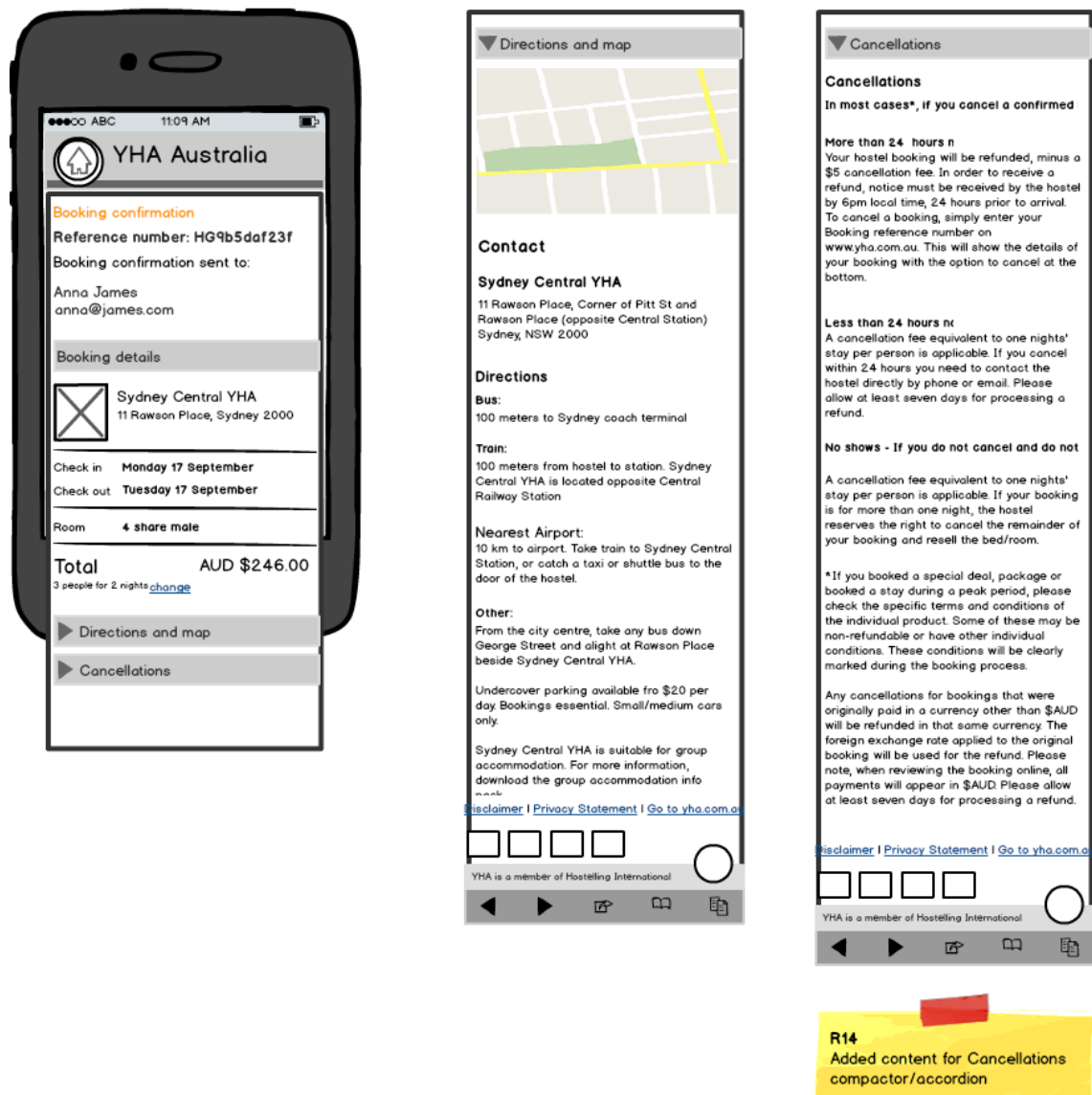
**R18**  
Minor changes to the Booking summary to make it more consistent across all 3 steps in process.

**R11**  
Added lightbox with content to Security



## C7 Confirmation

### YHA mobile wireframes - Confirmation - Page 9



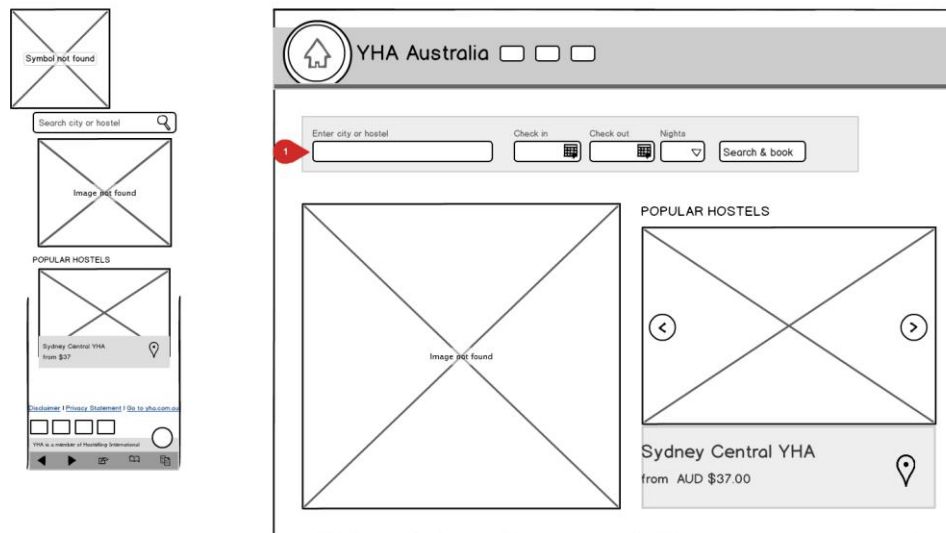


## Anexo D: Wireframes de escritorio

Los *wireframes* de escritorio se crearon a partir de los móviles, que sirven como base para documentar la estructura, progresión y funcionalidad. Los *wireframes* de escritorio describen la transición a la nueva resolución desde el los diseños móviles.

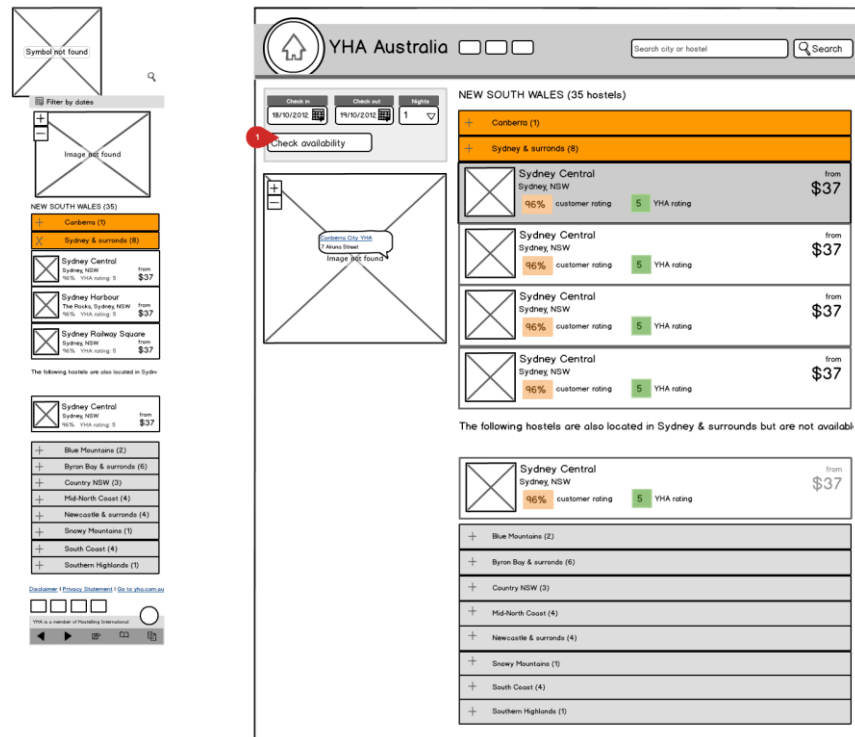
### D1 Home

YHA language desktop wireframes - Home



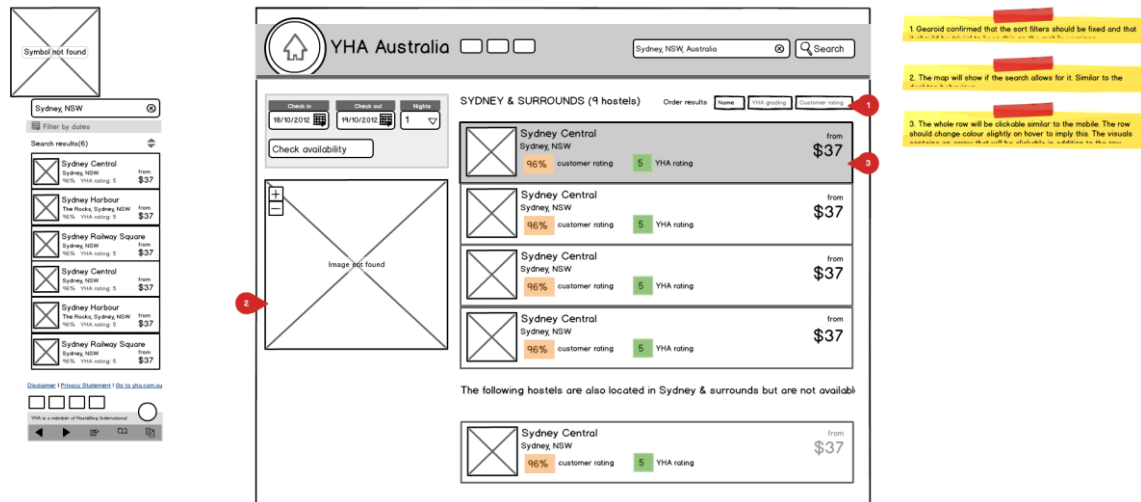
### D2 Map Search

YHA language desktop wireframes - Map search



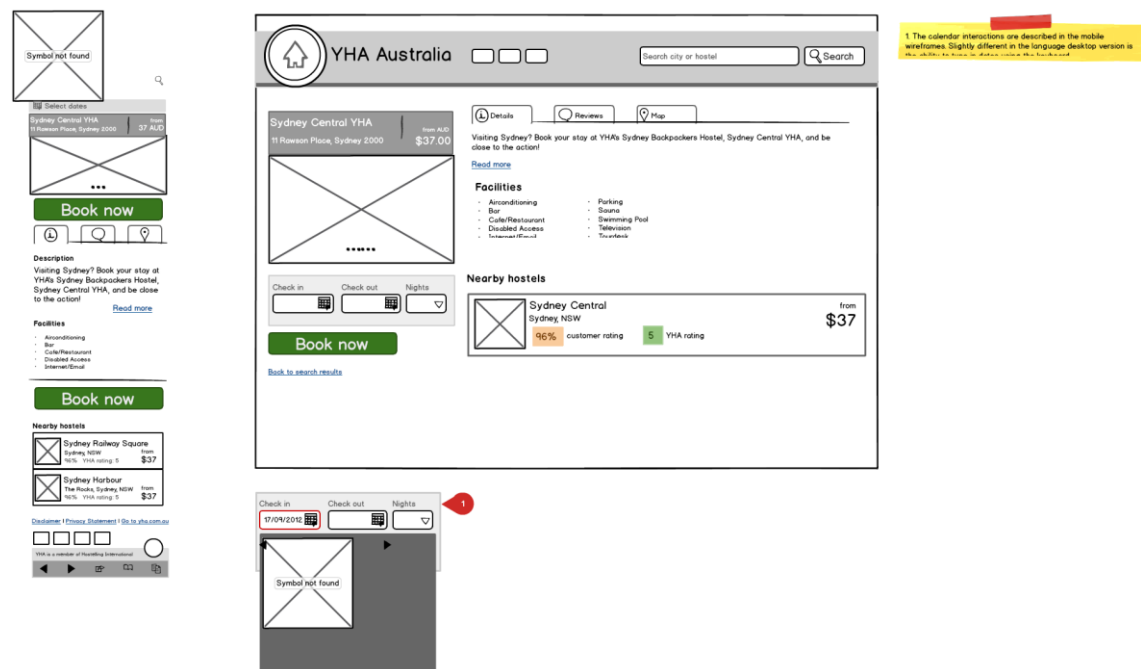
## D3 String Search

YHA language desktop wireframes - String search



## D4 Hostel details

YHA language desktop wireframes - Hostel details



## D5 Select Rooms

YHA language desktop wireframes - Select rooms

Symbol not found

1 Room type 2 3

Sydney Central YHA  
11 Rawson Place, Sydney 2000

Check in

Check out

Nights

MON SEPT 17

WED SEPT 19

2

4 share female  
per night from \$38

4 share female  
per night from \$38

4 share female  
per night from \$38

4 share female  
per night from \$38

4 share female  
per night from \$38

Looking for family rooms?

5 bed family room  
per room per night from \$100

Phone hostel to book

Not available for your dates

4 share female  
per night from \$38

Disclaimer | Privacy Statement | Go to yha.com.au

YHA is a member of Booking International

YHA Australia

Search city or hostel

Search

Sydney Central  
11 Rawson Place, Sydney 2000

Check in

Check out

Nights

1 Room type

2 Guests

3 Payment

Available rooms

4 share female  
per night from \$38

4 share female  
per night from \$38

4 share female  
per night from \$38

4 share female  
per night from \$38

Looking for family rooms?

5 bed family room  
per room per night from \$100

Phone hostel to book

Unavailable

4 share female  
per night from \$38

1 The progress bar placement differs from the mobile version

## D6 Select guests

YHA language desktop wireframes - Select guests

Symbol not found

1 2 Guests 3

Sydney Central YHA  
11 Rawson Place, Sydney 2000

Check in

Check out

Monday 17 September

Tuesday 17 September

Room

4 share male

Member rate

\$76.00 for 2 nights

1 guest

Standard rate

\$77.00 for 2 nights

2 guests

Total

AUD \$246.00

3 people for 2 nights

Go to payment

Note you can book member rates now and pay for a YHA

YHA Australia

Search city or hostel

Search

Sydney Central  
11 Rawson Place, Sydney 2000

Check in

Check out

Monday 17 September

Tuesday 17 September

Room

4 share male

1 Room type

2 Guests

3 Payment

YHA Member rate \$76.00 for 2 nights

YHA Standard rate \$77.00 for 2 nights

Guests

1

Guests

1

2 people for 2 nights

Total

\$138AUD

Go to payment


55

## YHA language desktop wireframes - Payment details


### D8 Confirmation

## YHA language desktop wireframes - Confirmation

56



# YHA Australia



## Sydney Central

11 Rawson Place, Sydney 2000

Phone number: +617 4051 0772

Reception hours: 24h

Check in	Monday 17 September
Check out	Tuesday 17 September
Room	4 share male

2 people for 2 nights      \$138AUD

[Change](#)

1 Room type
2 Guests
3 Payment

Guest 1

☒ Will

☒ Last name

☒ Gender

☒ Age

☒ Member

☒ Member no.

Guest 3

Guest 4

Main contact person

☒ Select guest

☒ Country of residence

☒ Email

☒ Confirm email

Arrival time

☒ Arrival time

Comments(optional)

Payment details

☒ Visa

☒ Card number

☒ Name on card

Expiration date  Security code

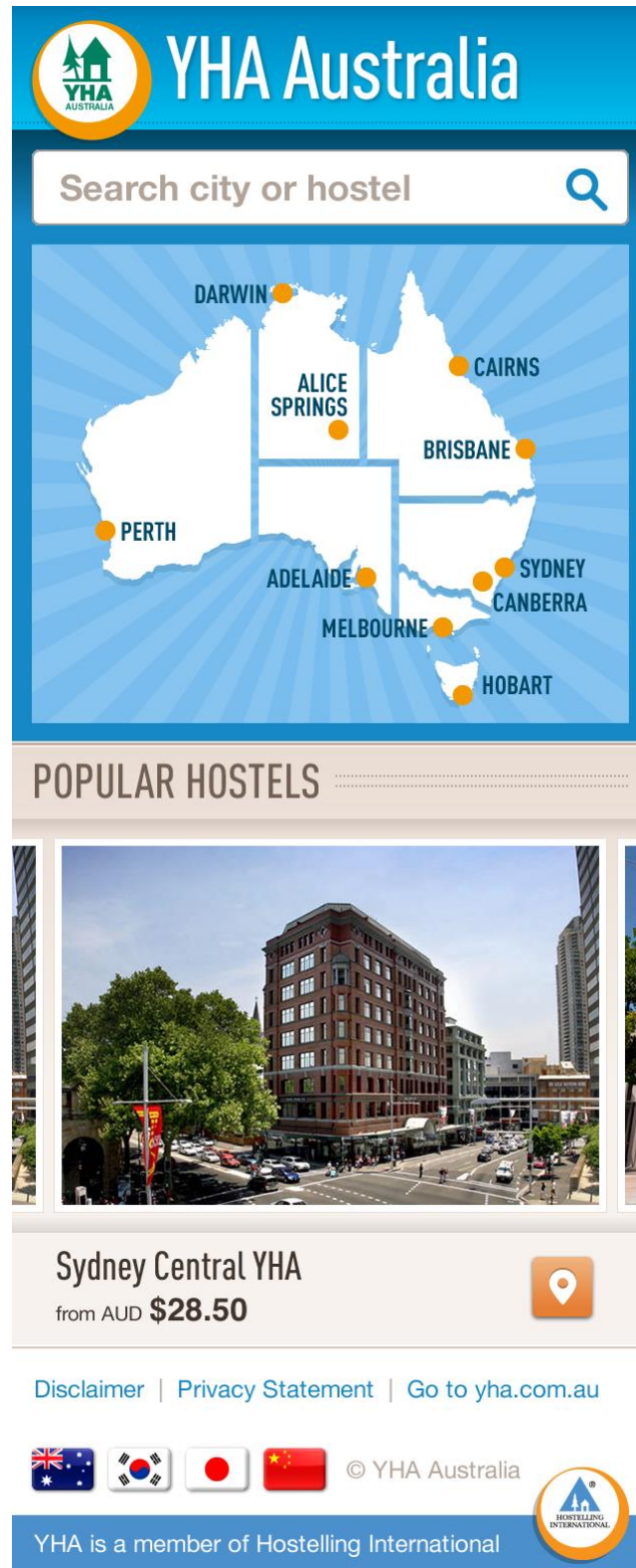
☒ MM/YY  ☒

☐ Yes, I have read the Booking Terms and Conditions for the hostel I am booking, and I am aware that a cancellation more than 24 hours in advance will incur a 15% cancellation fee, and a cancellation of less than 24 hours in advance a cancellation fee equivalent to one overnight stay per person.


## Anexo E: Diseños visuales móviles

Diseños visuales para el entorno móvil basados en los *wireframes* anteriores y preparados para ser utilizados en las plantillas.


### E1 Página de inicio




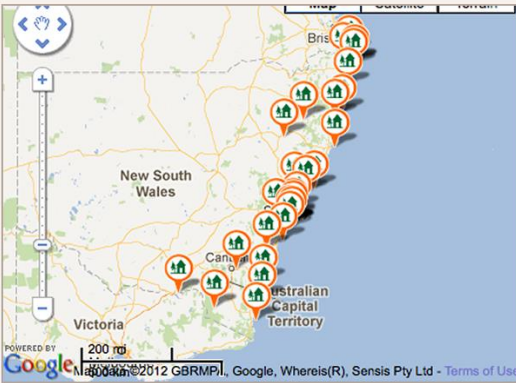
## E2 Resultados de búsqueda por localización



# YHA Australia



 Filter by dates



Map of New South Wales showing hostel locations. The map includes labels for New South Wales, Victoria, Canberra, Australian Capital Territory, and Sydney. A scale bar indicates 200 km. The map is powered by Google Maps.


## NEW SOUTH WALES (35 hostels)

- + Sydney & surrounds (9)
- + Canberra (1)


---

- + Blue Mountains (2)
- + Byron Bay & surrounds (6)
- + Country NSW (3)
- + Mid-North Coast (4)
- + Newcastle & surrounds (4)
- + Snowy Mountains (1)
- + South Coast (4)
- + Southern Highlands (1)


[Disclaimer](#) | [Privacy Statement](#) | [Go to yha.com.au](#)

 © YHA Australia


YHA is a member of Hostelling International




## E3 Resultados de búsqueda filtrados por disponibilidad




# YHA Australia



 14-16 Oct | 2 nights


### SYDNEY & SURROUNDS (9 hostels)



**Sydney Central**  
Sydney, NSW

90% YHA Grading: 5


from AUD **\$37<sup>00</sup>**



**Sydney Harbour - The Rocks**  
Sydney, NSW

96% YHA Grading: 5


from AUD **\$41<sup>00</sup>**



**Sydney, Bondi Beach**  
Bondi Beach, NSW

79% YHA Grading: 4


from AUD **\$30<sup>00</sup>**



**Sydney, Collaroy**  
Collaroy, NSW

87% YHA Grading: 4


from AUD **\$26<sup>00</sup>**



**Sydney, Cronulla**  
Cronulla, NSW

79% YHA Grading: 4


from AUD **\$30<sup>00</sup>**



**Sydney, Glebe**  
Glebe, NSW

86% YHA Grading: 4

from AUD **\$27<sup>00</sup>**




**Sydney, Pittwater**  
Church Point, NSW

96% YHA Grading: 3

from AUD **\$27<sup>00</sup>**


The following hostels are also located in Sydney & surrounds but are not available for your dates:



**Sydney Railway Square**  
Sydney, NSW

96% YHA Grading: 5

from AUD **\$29<sup>00</sup>**




**Sydney Summer Hostel**  
Chippendale, NSW


88% YHA Grading: 4

from AUD **\$85<sup>00</sup>**

[Disclaimer](#) | [Privacy Statement](#) | [Go to yha.com.au](#)



© YHA Australia

YHA is a member of  Hostelling International





YHA Australia

Sydney, NSW, Australia

31

Filter by dates

Check in

SUN OCT 14

Check out

TUE OCT 16

Nights

2

◀

OCTOBER 2012


▶

Sun	Mon	Tue	Wed	Thu	Fri	Sat
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3


clear dates




## E5 Detalles de hotel



# YHA Australia



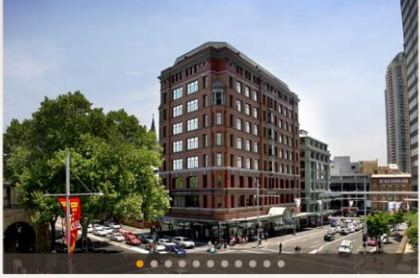


14-16 Oct 2 nights




## Sydney Central YHA

11 Rawson Place, Sydney 2000

from AUD **\$28<sup>50</sup>**



### Book now



### Description

Visiting Sydney? Book your stay at YHA's Sydney Backpackers Hostel, Sydney Central YHA, and be close to the action!


[Read more](#)

### Facilities

✓ Air conditioning	✓ Sauna
✓ Bar	✓ Swimming Pool
✓ Cafe/Restaurant	✓ Television
✓ Disabled access	✓ Tourdesk
✓ Internet/Email	✓ Meeting Room
✓ Parking	

### Book now


#### Nearby hostels



### Sydney Railway Square

Sydney, NSW

from AUD **\$29<sup>00</sup>**




### Sydney Harbour - The Rocks

Sydney, NSW


from AUD **\$41<sup>00</sup>**

[Disclaimer](#) | [Privacy Statement](#) | [Go to yha.com.au](#)



© YHA Australia

YHA is a member of Hostelling International



## E6 Selección de habitación



1. ROOM TYPE2.3.



**Sydney Central YHA**  
11 Rawson Place,  
(opposite Central Station),  
Sydney 2000

Check in

SUN OCT 14

Check out

TUE OCT 16

Nights

2



4 share female  
per night from  
**\$38<sup>00</sup>**



4 share male  
per night from  
**\$38<sup>00</sup>**



4 share mixed  
per night from  
**\$38<sup>00</sup>**



6 share female  
per night from  
**\$28<sup>50</sup>** **Limited availability**



Double room  
per room per night from  
**\$95<sup>00</sup>** **Minimum 2 night stay**



Twin room  
per room per night from  
**\$95<sup>00</sup>** **Maximum 7 night stay**

**Looking for family rooms?**



5 Bed Family Room  
per room per night from \$130  
 [Phone hostel to book](#)

**Not available for your dates:**



6 share male  
per night from  
**\$28<sup>50</sup>** **Not available**




Double/twin private bathroom  
per room per night from  
**\$110<sup>00</sup>** **Minimum 5 night stay**

[Disclaimer](#) | [Privacy Statement](#) | [Go to yha.com.au](#)

 © YHA Australia 

YHA is a member of Hostelling International


## E7 Selección de huéspedes



1.

2. GUESTS

3.



**Sydney Central YHA**  
11 Rawson Place,  
(opposite Central Station),  
Sydney 2000

Check in

**Sunday 14 October**

Check out

**Tuesday 16 October**

Room

**4 share male**

**Member rate**  
\$76.00 for 2 nights

1 guest

**Standard rate**  
\$170.00 for 2 nights

2 guests


**Total**  
3 people for 2 nights

**AUD \$246<sup>00</sup>**


**Go to payment**

**Note:** you can book member rates now and pay for a YHA membership when you arrive at the hostel.

## E8 Detalles de huésped y pago



1. > 2. > 3. PAYMENT



**Sydney Central YHA**  
11 Rawson Place,  
(opposite Central Station),  
Sydney 2000

Check in

**Sunday 14 October**

Check out

**Tuesday 16 October**

Room

**4 share male**

3 people for 2 nights

**AUD \$246<sup>00</sup>**

[change](#)

Guest 1

✓ Brett

✓ Vaughan

✓ Gender

✓ Age

✓ YHA member

✓ Member number

Guest 2

Guest 3

Main contact person

✓ Select guest

✓ Country of residence

✓ Email

✓ Confirm email

Arrival details

✓ Arrival time

Comments (optional)

Payment details

✓ Visa

✓ Card number

✓ Name on card

Expiry date

Security code


✓ MM/YY

✓

☐ Yes, I have read the [Booking Terms and Conditions](#) for the hostel I am booking, and I am aware that a cancellation more than 24 hours in advance will incur a 5% cancellation fee, and a cancellation of less than 24 hours in advance will incur a cancellation fee equivalent to one overnight stay per person.

Confirm payment


## E9 Confirmación de la reserva



**YHA Australia**

BOOKING CONFIRMATION

Reference number: [HG9b5daf23f](#)

Booking confirmation sent to:  
Brett Vaughan  
[brett@vaughan.co.nz](mailto:brett@vaughan.co.nz)



 **Booking details**


**Sydney Central YHA**  
 **+612 9218 9000**

Check in **Sunday 14 October**  
Check out **Tuesday 16 October**


4 share male **\$137.00**  
3 people for 2 nights


**AMOUNT PAID** **\$137.00**

 **Directions and map** 



**Sydney Central YHA**  
11 Rawson Place,  
Corner of Pitt St and Rawson Place  
(opposite Central Station)  
Sydney, NSW 2000

 **Call (+612) 9218 9000**

 **[sydcentral@yha.com.au](mailto:sydcentral@yha.com.au)**

**Reception hours:**  
24 hour reception. Check-in from 1pm.



**Directions**

**Bus:**  
100 metres to Sydney coach terminal.



**Train:**  
100 metres from hostel to station. Sydney Central YHA is located opposite Central Railway Station.


**Nearest airport:**  
10 km to airport. Take train to Sydney Central Station, or catch taxi or shuttle bus to the door of the hostel.

**Other:**  
From the city centre, take any bus down George Street and alight at Rawson Place beside Sydney Central YHA.  
  
Undercover parking available for \$20 per day. Bookings essential. Small/medium cars only.

 **Cancellations** 

[Disclaimer](#) | [Privacy Statement](#) | [Go to yha.com.au](#)

 © YHA Australia 

YHA is a member of Hostelling International 

## Anexo F: Ejemplos visuales de resolución de escritorio y lenguaje japonés

Siguiendo el patrón de los *wireframes* para la resolución de escritorio solo se produjeron dos pantallas de muestra para el cliente. Los estilos visuales de escritorio se derivan de las pantallas móviles.


### F1 Página de inicio




## F2 Resultados de búsqueda por localización en lenguaje japonés




# YHA オーストラリア



31 日付による絞り込み

検索街またはホステル 



 シドニーとその周辺 (9)


 キャンベラ (1)




カンベラ市 YHA  
Canberra City YHA  
カンベラ市 ACT


AUD에서  
**\$28<sup>50</sup>**

 ブルーマウンテンズ (2)


 バイロンベイとその周辺 (6)

 カントリーサイド (3)

 ミッドノースコースト (4)

 ニューカッスルと囲む (4)


 スノーウィー・マウンテンズ (1)

 南岸 (4)

 サザンハイランド (1)

免責事項 | プライバシーステートメント | に行く [yha.com.au](http://yha.com.au)

© YHA オーストラリア



YHA ホステリング・インターナショナルのメンバーです

67

## Anexo G: Distribución de horas de trabajo mensuales

Horas de desarrollo *fron end* empleadas por el autor del proyecto.

	Septiembre 2012	Octubre 2012	Noviembre 2012	Diciembre 2012	
Horas de trabajo	36.00	137.00	144.50	91.00	
Total					408.50





## Referencias

- [1] YHA Australia, 2014. Página oficial YHA Australia. <http://www.yha.com.au/>
- [2] Hostelling International, 2014. Página oficial HI. <https://www.hihostels.com/>
- [3] Jakob Nielsen, 2009. Página oficial de Nielsen Norman Group. <http://www.nngroup.com/articles/mobile-usability-first-findings/>
- [4] Daniel Roberts, 2010. Página oficial de Usability.gov. <http://www.usability.gov/get-involved/blog/2010/05/mobile-device-usability.html>
- [5] A List Apart, 2014. Página oficial A List Apart. <http://alistapart.com/>
- [6] Ethan Marcotte, 2009. Página oficial A List Apart. <http://alistapart.com/article/fluidgrids>
- [7] Ethan Marcotte, 2010. Página oficial A List Apart. <http://alistapart.com/article/responsive-web-design>
- [8] W3C, 2012. Página oficial W3C. <http://www.w3.org/TR/css3-mediaqueries/>
- [9] W3C, 2014. Página oficial W3C. <http://www.w3.org/Style/CSS/Overview.en.html>
- [10] ISO Standard 9241-210:2010. Página oficial ISO. [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=52075](http://www.iso.org/iso/catalogue_detail.htm?csnumber=52075)
- [11] Google Maps para desarrolladores, 2014. Página oficial Google Maps. <https://developers.google.com/maps/>
- [12] Steve Souders, 2007. Página oficial Yahoo Developer Network. <https://developer.yahoo.com/blogs/ydn/high-performance-sites-rule-6-move-scripts-bottom-7200.html>
- [13] Addy Osmani (2014). *Learning JavaScript Design Patterns* Online Edition. ISBN: 978-1-4493-3486-4  
<http://addyosmani.com/resources/essentialjsdesignpatterns/book/#modulepatternJavaScript>
- [14] .Net Framework 4, 2007. Página oficial MSDN Library. [http://msdn.microsoft.com/en-us/library/vstudio/bb399001\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/vstudio/bb399001(v=vs.100).aspx)
- [15] Embedded JavaScript does not work after async postback from UpdatePanel. Respuesta en Stack Overflow. <http://stackoverflow.com/questions/8092728/embedded-javascript-does-not-work-after-async-postback-from-updatepanel>
- [16] Hajan Selmani, 2010. Weblog personal de Hajan Selmani <http://weblogs.asp.net/hajan/make-them-to-love-each-other-asp-net-ajax-updatepanels-and-javascript-jquery-functions>
- [17] W3C, 2008. Página oficial W3C. <http://www.w3.org/wiki/CSS/Properties/float>

- [18] W3C, 2011. Página oficial W3C. <http://www.w3.org/TR/2011/WD-html5-20110525/forms.html#the-form-element>
- [19] Plugin jQuery Autocomplete. Página oficial jQuery UI <http://jqueryui.com/autocomplete/>
- [20] W3C, 2014. Página oficial W3C. <http://www.w3.org/TR/geolocation-API/>
- [21] Plugin Moment. Página oficial Moment.js <http://momentjs.com/>
- [22] Plugin Markerclusterer v3. <http://google-maps-utility-library-v3.googlecode.com/svn/trunk/markerclusterer/docs/reference.html>
- [23] Plugin h5validate. <http://ericleads.com/h5validate/>
- [24] Graceful degradation. Página oficial de Wikipedia [http://en.wikipedia.org/wiki/Fault\\_tolerance](http://en.wikipedia.org/wiki/Fault_tolerance)
- [25] Plugin mediaqueries.js. <https://code.google.com/p/css3-mediaqueries-js/>
- [26] Plugin html5shiv. <https://code.google.com/p/html5shiv/>
- [27] Richard Clark, 2009. Página oficial HTML5 Doctor. <http://html5doctor.com/html-5-reset-stylesheet/>
- [28] W3C, 2011. Página oficial W3C. <http://www.w3.org/TR/CSS21/visuren.html>
- [29] Servidor BoomShed (root directory empty) <http://boomshed.com/>
- [30] W3C, 2013. Página oficial W3C. <http://www.w3.org/TR/css-transforms-1/>
- [31] Increase Your Site's Performance with Hardware-Accelerated CSS. <http://blog.teamtreehouse.com/increase-your-sites-performance-with-hardware-accelerated-css>
- [32] ANDROID BROWSER: Page scrolls to bottom when user focuses or types in an INPUT field that has parent with CSS value: -webkit-backface-visibility: hidden; <https://code.google.com/p/android/issues/detail?id=13048>
- [33] Content Delivery Network. Página oficial Wikipedia [http://en.wikipedia.org/wiki/Content\\_delivery\\_network](http://en.wikipedia.org/wiki/Content_delivery_network)
- [34] Window.matchMedia(), 2014. Pagina oficial Mozilla Developer Network <https://developer.mozilla.org/en-US/docs/Web/API/Window.matchMedia>