

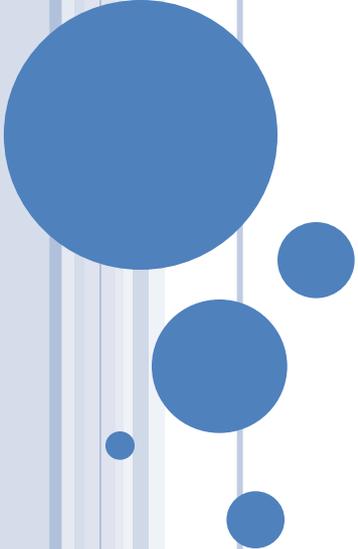
Área **J2EE**

Autor: **Manuel Luis Ortiz Rojas (ITIG)**

Coordinador: **Juan José Rodríguez**

Universitat Oberta de Catalunya

Junio 2012



TRABAJO FIN DE CARRERA

EL CUADERNO DEL ASTRÓNOMO

ÍNDICE

1. Introducción	página 3
2. Enfoque y planificación	página 5
• Ciclo de vida	página 5
• Planificación	página 6
• Productos Obtenidos	página 7
3. Análisis	página 8
• Actores	página 8
• Subsistemas de registro y conexión.	página 9
• Subsistemas de gestión de equipo y lugares	página 10
• Subsistemas de gestión de observaciones y amigos	página 11
4. Diseño	página 12
• JBoss Seam	página 12
• Diagrama de clases	página 13
• Modelo de datos	página 14
5. Implementación	página 15
• Seam-gen	página 15
• Modelo, vista y controlador	página 16
6. Conclusiones	página 17



1. INTRODUCCIÓN

○ Justificación y contexto

- Aumentar y mejorar la formación en J2EE.
- Afianzar conocimientos de otras asignaturas.

○ Objetivos

- Aportar una herramienta útil a la comunidad de astrónomos aficionados. Para ello se establecen una serie de hitos planificados en el tiempo que se deberán cumplir para garantizar el éxito del desarrollo.

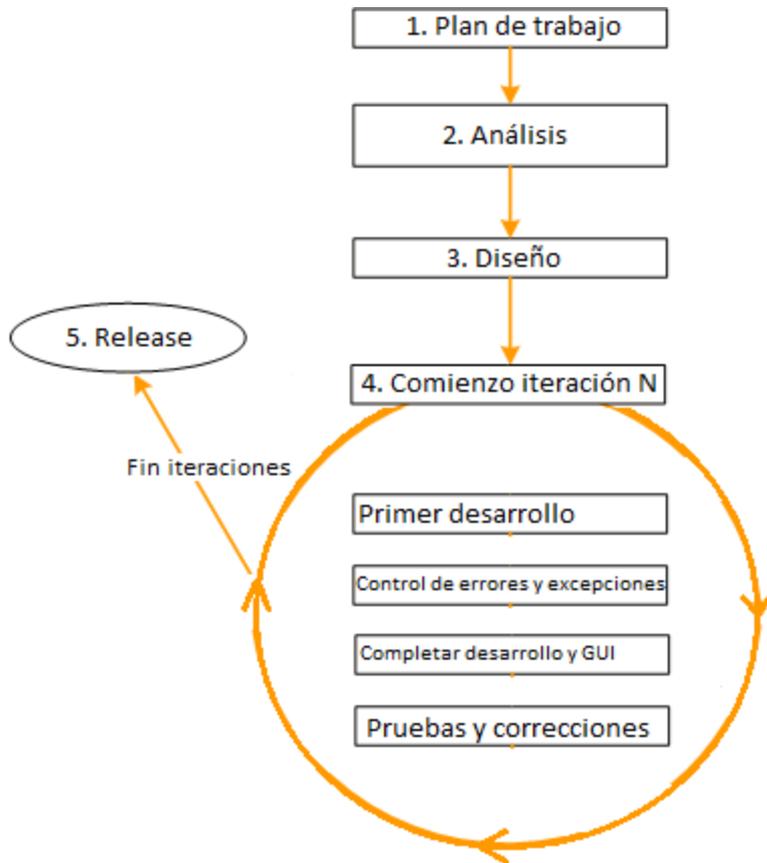


Hitos marcados a lo largo del desarrollo

- **Plan de Trabajo:** con el objetivo de establecer una descripción y una planificación realista del proyecto a desarrollar
- **Análisis y diseño:** documentación necesaria para sentar la base de implementación del proyecto.
- **Implementación:** entregable de la aplicación a desarrollar y manual de instalación .
- **Documentación final:** memoria y presentación.



2. ENFOQUE Y PLANIFICACIÓN



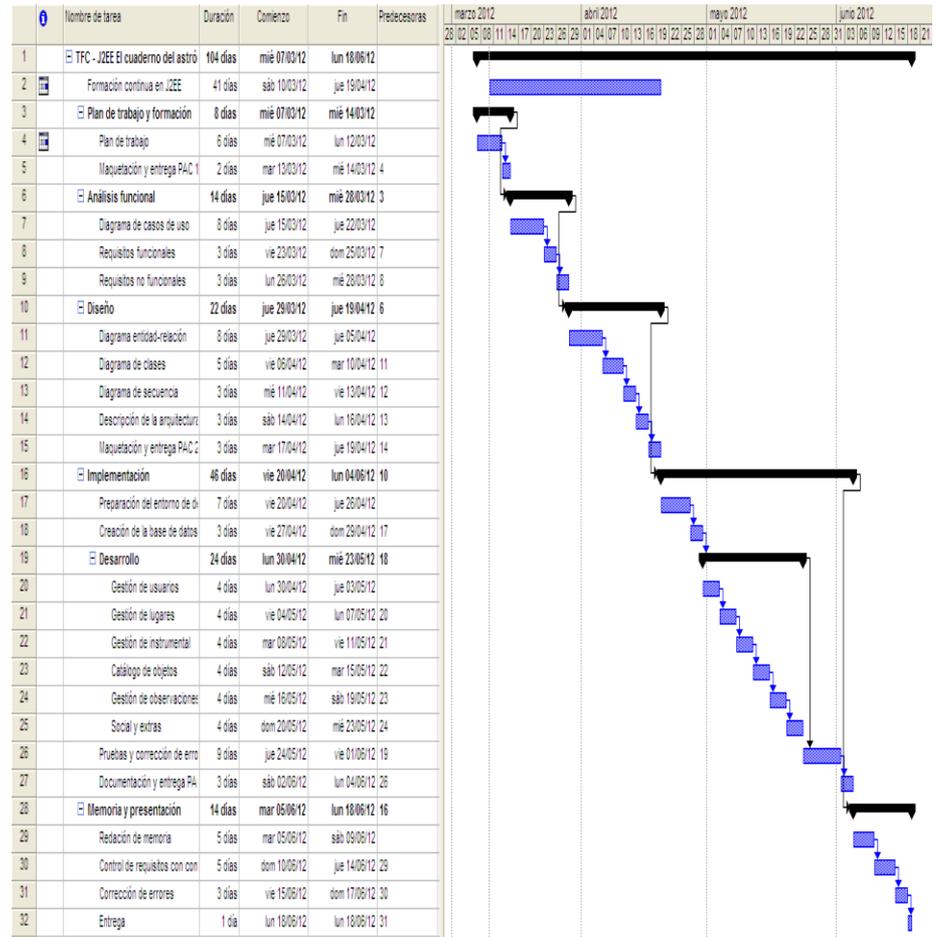
Ciclo de vida iterativo

Cada caso de uso pasará por varias iteraciones hasta llegar a su versión final.

De esta forma nos acercamos más a metodologías ágiles como SCRUM para poder ajustarnos a las necesidades reales del proyecto; ya que en cada iteración se realiza una entrega al cliente y podemos ajustar el desarrollo a tiempo.

○ Planificación

- Identificación de objetivos.
- Distribución en el tiempo según carga de trabajo y conocimientos propios.
- Formación continua a lo largo de todo el desarrollo.



Productos obtenidos:

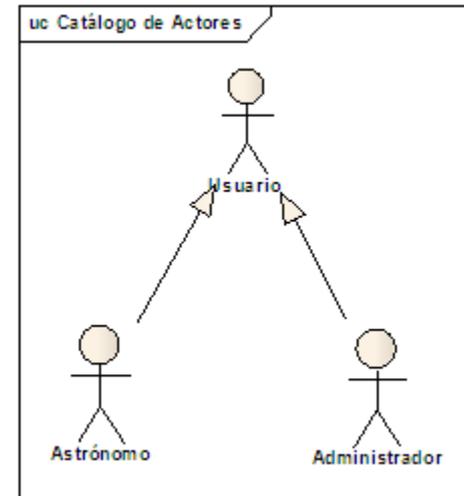
- **Plan de trabajo:** en esta fase se han establecido los objetivos que deberá cumplir el proyecto, así como una planificación de todo el trabajo a realizar teniendo en cuenta los recursos disponibles, formación y tiempo.
- **Análisis y diseño:** en este documento encontraremos la siguiente información:
 - Casos de uso por subsistema.
 - Diseño y arquitectura.
 - Prototipo gráfico.
- **Implementación:** fase en la que se ha desarrollado el producto y además se ha generado la documentación para su correcta instalación y puesta en funcionamiento. La entrega consta principalmente de un fichero EAR y el script de base de datos.
- **Entrega final y memoria:**
 - Producto final: entrega definitiva del producto tal y como se ha descrito en el punto anterior.
 - Memoria: formada por este documento el cual sintetiza toda la información generada durante el transcurso de cada una de las fases de este trabajo.
 - Presentación: documento en forma de diapositivas el cual sintetizará todo el trabajo realizado.



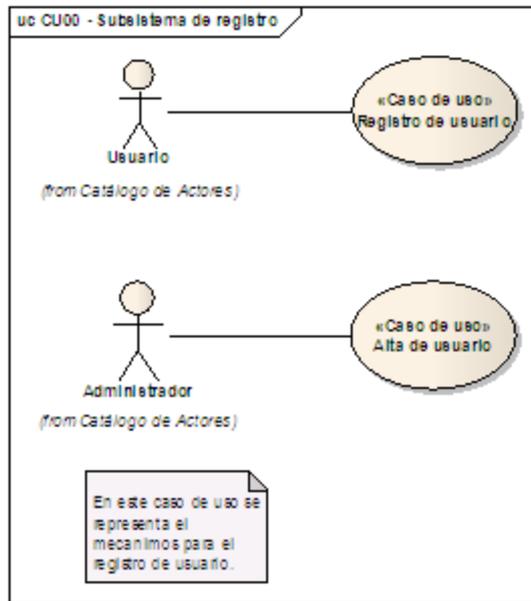
3. ANÁLISIS

Actores

- Se identifican los actores que intervienen en la aplicación.
- Se establece el rol de cada uno de ellos, funcionalidades disponibles y limitaciones.

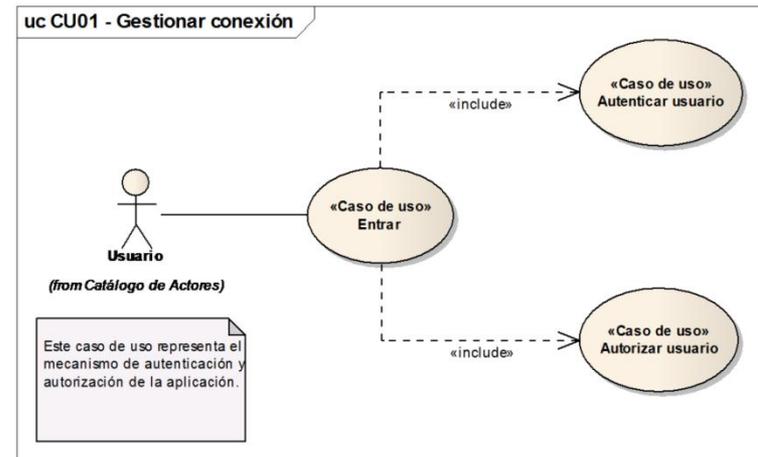


○ Subsistema de registro



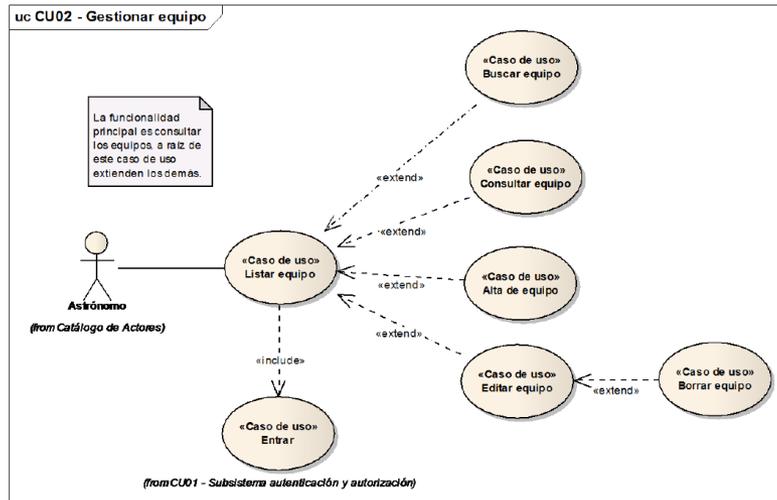
- Contempla el registro de usuarios con rol de astrónomo de forma pública.
- Alta de usuarios en el sistema por parte del administrador.

○ Subsistema de conexión



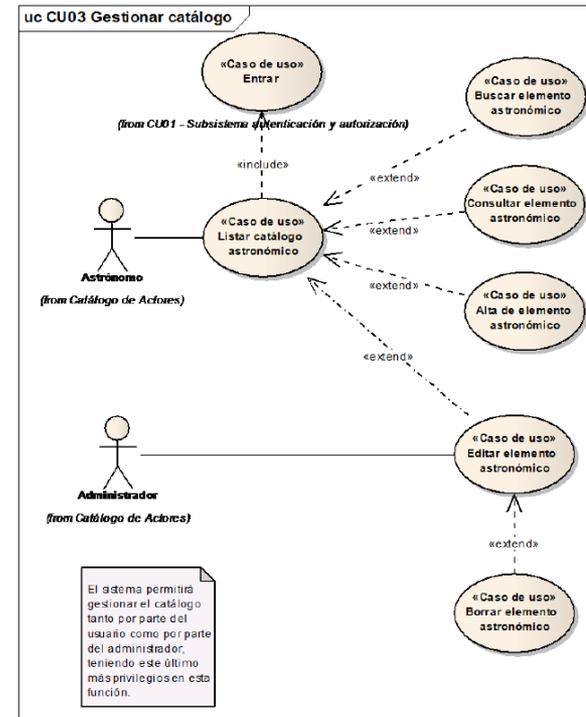
- Autentica al usuario comprobando sus credenciales contra la base de datos. La contraseña está cifrada en MD5.
- Autoriza al usuario con el perfil que le corresponde.

○ Gestión de equipo



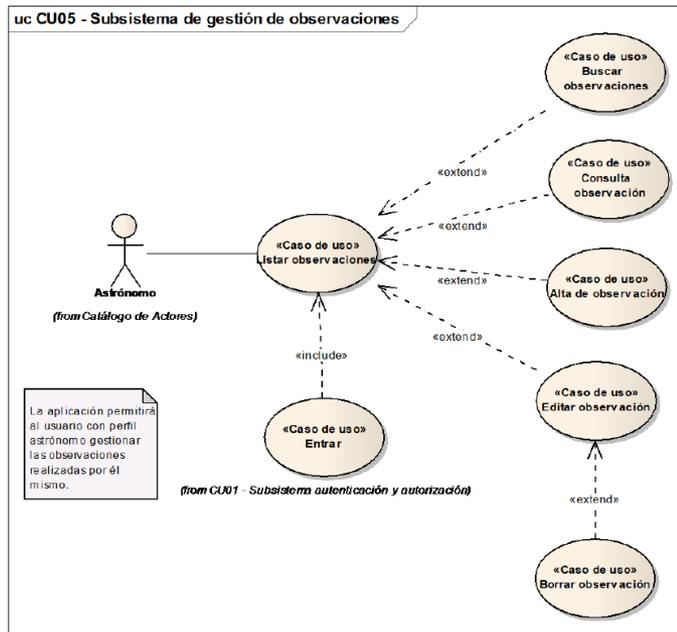
- Gestión personal del equipo.
- Alta de material de observación y fotografía por categorías.
- Edición, borrado y baja disponibles.

○ Gestión de lugares

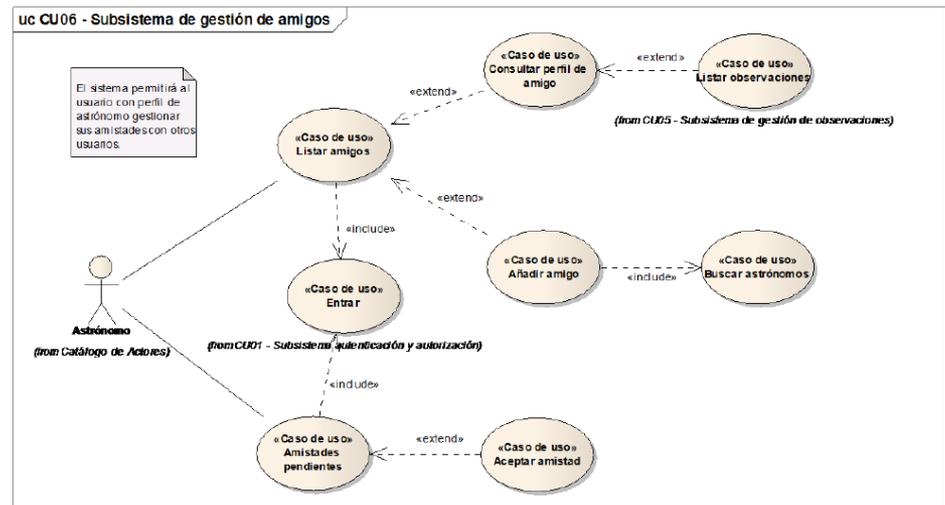


- Gestión personal de los lugares de observación.
- Edición, borrado y baja disponibles.
- Uso de la API de Google Maps.

○ Gestión de observaciones



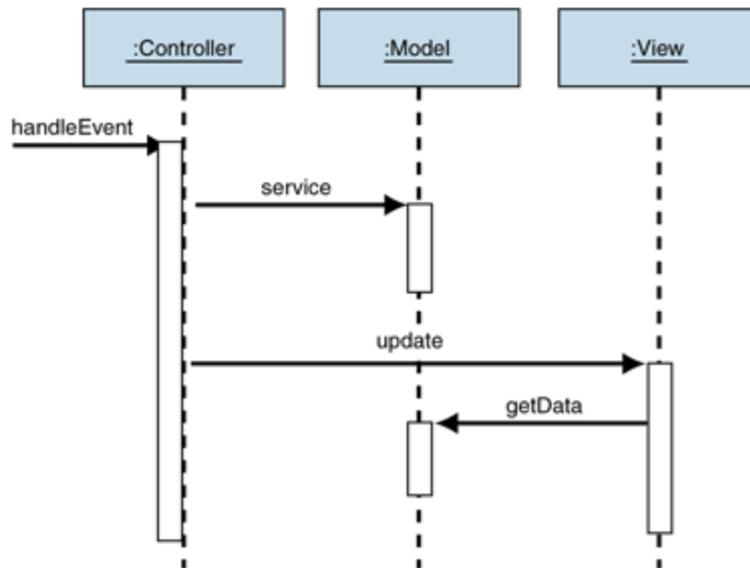
○ Gestión de amigos



- Gestión personal de amigos.
- Envío de solicitudes de amistad.
- Aceptación de solicitud de amistad pendiente.
- Consulta de las observaciones de un amigo

4. DISEÑO

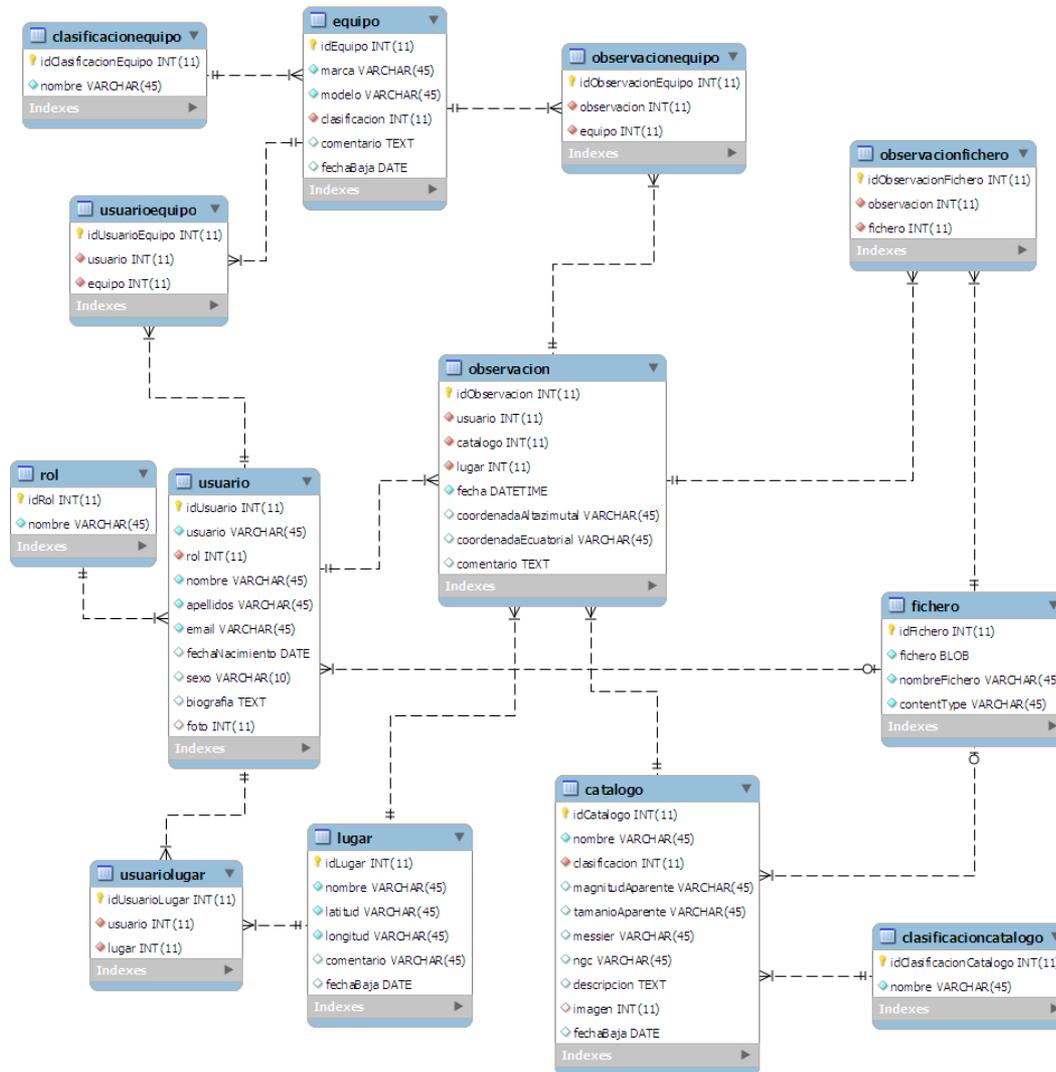
- JBoss Seam coge lo mejor de EJB3 y JSF añadiendo además herramientas muy útiles como Richfaces para la interfaz o seam-gen para la generación de código.



MVC

- **Controlador:** basado en las clases *EntityHome*, *EntityQuery* y *EntityController*.
- **Modelo:** basado en EJB3 y anotaciones con Hibernate para el mapeo objeto-relacional.
- **Vista:** basada en ficheros XHTML que hacen uso de librerías como a4j, JSF, Richfaces y otras no incluidas como ICEfaces.

Modelo de datos



- Construcción del modelo de datos para MySQL 5.

- ID auto incrementables.



○ Modelo

Beans de entidad con mapeo objeto – relacional realizado con *Hibernate Annotations*.

Estas clases están contenidas en el paquete:

uoc.edu.cuadernoastrologo.entity

Ejemplo de mapeo de una tabla:

```
@Entity
@Table(name = "lugar", catalog = "cuadernoastrologo")
public class Lugar implements java.io.Serializable {
```

Mapeo de una columna y un atributo:

```
@Column(name = "longitud", nullable = false, length = 45)
@NotNull
@Length(max = 45)
public String getLongitud() {
    return this.longitud;
}
```

○ Vista

Las vistas se han realizado principalmente combinando JSF y Richfaces.

En algunos puntos se ha hecho uso de la API de gmaps4JSF y a4j.

Ejemplo de uso de gmap4jsf en el fichero Lugar.xhtml

```
<m:map width="500px" height="400px"
latitude="#{lugarHome.instance.latitud}"
longitude="#{lugarHome.instance.longitud}"
<m:mapControl name="GLargeMapControl"
position="G_ANCHOR_BOTTOM_RIGHT"/>
<m:mapControl name="GMapTypeControl"/>
<m:marker latitude="#{lugarHome.instance.latitud}"
longitude="#{lugarHome.instance.longitud}" />

<m:htmlInformationWindow
latitude="#{lugarHome.instance.latitud}"
longitude="#{lugarHome.instance.longitud}"
htmlText="#{lugarHome.instance.nombre}" />

</m:map>
```

○ Controlador

Las clases *Home* son las encargadas de gestionar la persistencia de la aplicación a través del *EntityManager* de Seam.

Las clases *List* y *Controller* son usadas para realizar las búsquedas y el resto de consultas a la base de datos.

El flujo de la aplicación es controlado a través de los mensajes devueltos por los métodos de estas clases y las reglas de navegación establecidas en los ficheros **.page.xml**.



6. CONCLUSIONES

- **Importancia de la planificación**
 - Estimación realista de cada tarea.
 - Dejar siempre un tiempo para imprevistos, surgirán.
 - La formación también se ha de tener en cuenta.
- **El análisis y el diseño son la base de todo proyecto**
 - Es vital documentar bien esta fase, ya que será la guía del desarrollador.
 - Cambios en el diseño en la fase de implementación supondrán un retraso en la estimación inicial.
- **Eligiendo las herramientas adecuadas**
 - Cada proyecto tiene sus necesidades, es importante valorarlas adecuadamente y ajustarlas a las soluciones que el mercado ofrece.

