

*LINS - In.Co - Facultad de Ingeniería - Udelar*

# GUÍA DE RECURSOS DEL MIDES

*Informe Final – Proyecto de Grado - 2013-2014*

**INTEGRANTES:**

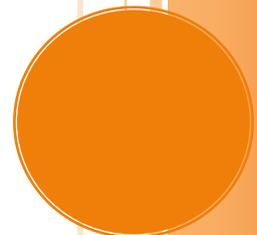
*Maximiliano Arcia*

*Germán Carlosena*

**TUTORAS:**

*Laura González*

*Raquel Sosa*





## Resumen

Debido a la cantidad de recursos y programas sociales que existen en el Uruguay, el Ministerio de Desarrollo Social (MIDES) decidió ofrecer una página web en la que se centraliza la información de los recursos y programas sociales que se llevan a cabo en el País. Esta página es denominada Guía de Recursos (GDR) y su finalidad es brindar una herramienta de búsqueda a la población sobre recursos y programas sociales existentes.

Desde su puesta en marcha, se comprobó que la GDR es una herramienta muy utilizada. Tiene un promedio de visitas que varían entre 2000 y 3000 semanalmente. Sin embargo, actualmente la guía tiene algunas problemáticas relacionadas principalmente a la actualización de la información y a los mecanismos de búsquedas que brinda.

Este proyecto consistió en el diseño y construcción de un sistema que aborda las problemáticas que actualmente tiene la GDR.

Durante el proyecto se realizó un estudio del marco conceptual enfocado al análisis de la realidad que tiene la Guía de Recursos y las mejoras que se deben incorporar. Dadas las características del contexto social del proyecto, se generaron instancias para intercambiar ideas y material con los funcionarios del MIDES responsables del mantenimiento de la GDR.

De estas reuniones surgieron los requerimientos del sistema, así como opciones tecnológicas a utilizar, como por ejemplo el desarrollo de una interfaz gráfica optimizada para dispositivos móviles, utilización de mapas y un nuevo mecanismo de actualización de recursos. Una de las funcionalidades más destacadas, es la desarrollada con el fin detectar de forma automática cambios en páginas web desde las que se toma información de los recursos.

Para el desarrollo de la solución tecnológica se trabajó con la plataforma *Java Empresarial*, se utilizó JBoss AS como servidor de aplicaciones y PostgreSQL como motor de base de datos. Además se utilizaron diversas Librerías, APIs y Frameworks como por ejemplo: Primefaces, Apache Shiro, JSoup, Google Maps, entre otras.

**Palabras Claves:** Recursos Sociales, Programas Sociales, Ministerio de Desarrollo Social (MIDES).

## TABLA DE CONTENIDOS

1	Introducción.....	6
1.1	Motivación .....	7
1.2	Objetivos .....	8
1.3	Aportes.....	9
1.4	Organización del documento .....	10
2	Marco Conceptual .....	12
2.1	Guía de Recursos del MIDES .....	12
2.2	Tecnologías para desarrollo empresarial.....	14
2.2.1	Plataformas de desarrollo empresarial.....	14
2.2.2	Tecnologías para Información Geográfica.....	17
2.2.3	Web Services .....	18
2.3	Desarrollo Dispositivos Móviles.....	19
2.3.1	Aplicaciones para dispositivos móviles.....	19
2.3.2	Interfaz web para dispositivos móviles.....	20
2.4	Conceptos de Seguridad.....	21
3	Análisis.....	24
3.1	Requerimientos .....	24
3.1.1	Relevamiento de Requerimientos .....	24
3.1.2	Modelo Dominio.....	27
3.2	Sistemas informáticos con objetivos similares.....	28
3.2.1	Sistema de Brasil .....	28
3.2.2	Sistema de Chile .....	30
3.2.3	Sistema de Argentina.....	33
3.2.4	Conclusiones.....	36
3.3	Casos de Uso.....	38
4	Solución Planteada .....	42
4.1	Arquitectura .....	42
4.2	Principales Funcionalidades.....	46
4.2.1	Detección automática de cambios en la web .....	46
4.2.2	Mecanismo de actualización de recursos.....	48
4.2.3	Visualización de la información geográfica .....	49
4.2.4	Búsqueda de Recursos .....	51
4.2.5	Gestión de Categorías .....	52
4.2.6	Dispositivos Móviles .....	53
4.2.7	Autenticación y Autorización .....	55
4.2.8	Hashing de contraseñas .....	56

4.3	Decisiones tomadas.....	58
4.3.1	Realizar una interfaz web mobile en vez de una aplicación mobile. ....	58
4.3.2	La implementación de ubicación geográfica utilizando Google Maps y no integrando con el sistema mapas que maneja el MIDES. ....	58
4.3.3	Usar parser Jsoup para la detección de cambios .....	58
4.3.4	Usar distancia de Levenshtein para búsqueda de texto.....	59
4.4	Problemas encontrados .....	60
4.4.1	Versión de JSF.....	60
4.4.2	Driver JDBC del servidor de OpenShift .....	60
4.4.3	Framework PrimeFaces para Desktop .....	60
4.4.4	Framework PrimeFaces para Mobile .....	60
4.4.5	Framework de seguridad Apache Shiro.....	60
5	Gestión de Proyecto.....	62
5.1	Etapas del Proyecto.....	62
5.1.1	Etapa 1 – Inicio del Proyecto y Análisis Preliminar .....	62
5.1.2	Etapa 2 – Investigación Tecnológica y Bosquejos de Interfaz .....	63
5.1.3	Etapa 3 –Elaboración y Construcción del Producto.....	64
5.1.4	Etapa 4 – Presentación de Avance y Mejora del Producto .....	64
5.1.5	Etapa 5 – Cierre del Proyecto, Documentación, y Presentación Final.....	64
5.2	Desvíos y dificultades .....	65
6	Validación con Usuarios.....	68
6.1	Herramienta utilizada.....	68
6.2	Algunos bosquejos .....	68
6.3	Resultado de la validación .....	72
7	Conclusiones y Trabajos a Futuro .....	74
7.1	Conclusiones .....	74
7.2	Trabajos a futuro .....	75
8	Referencias .....	76



# 1 INTRODUCCIÓN

La Guía de Recursos del MIDES (GDR) es una herramienta que tiene el propósito de brindar información sobre servicios, programas, centros de atención e instituciones, ya sean gubernamentales o no gubernamentales. Es una herramienta concebida con el doble propósito de servir de apoyo a operadores y técnicos del área social, y a la autogestión de información por parte de los ciudadanos. [1]

La GDR actualmente es un sistema web que ofrece dicha información, brindando además funcionalidades de búsqueda de la información disponible. A modo de ejemplo, si un ciudadano desea saber qué programas sociales existen para la educación en los jóvenes, éste podrá acceder al menú como se aprecia en la Figura 1.1

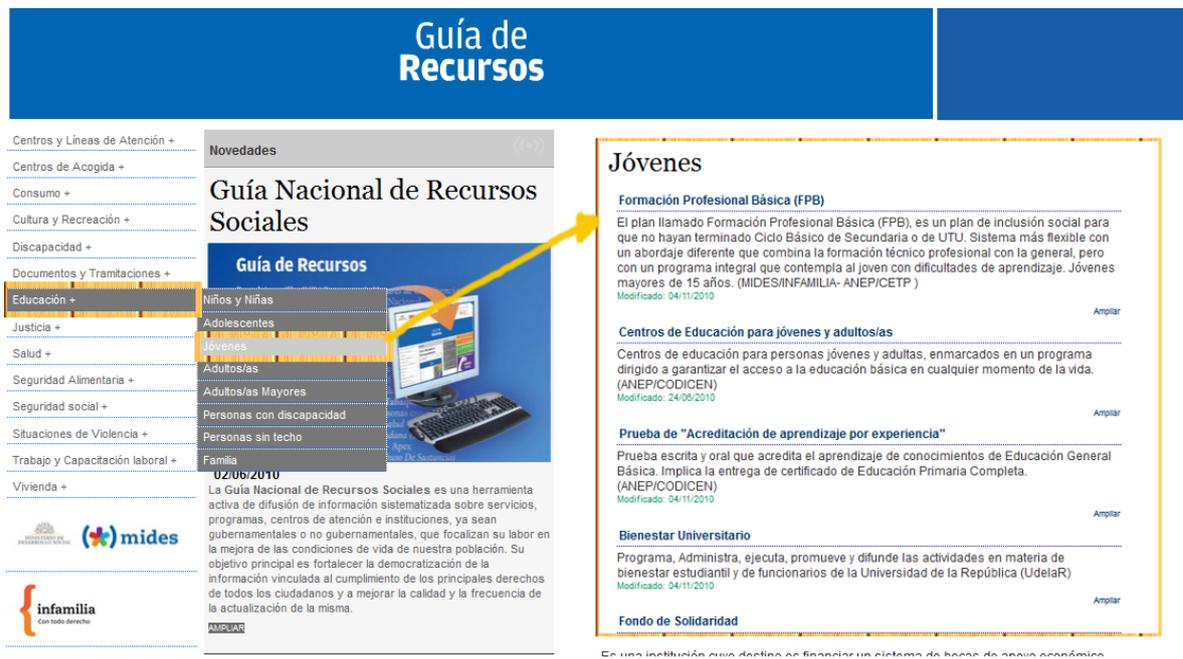


Figura 1.1 – Versión anterior de la GDR

De acuerdo a la información que brinda la GDR, es importante que se mantenga lo más actualizada posible y así tener certeza de que la información disponible, es válida al momento de consultarla.

## 1.1 Motivación

El soporte informático actual de la GDR, está basado en un Sistema de Gestión de Contenido Web.

*"La Guía de Recursos está construida sobre el producto InnovaPortal. La implementación de la guía sobre dicho producto se basa en el concepto de Contenedor. [2]*

*Esto es, los menús que categorizan los recursos son Contenedores que contienen a otros Contenedores (los recursos). Toda esta información es gestionada manualmente desde el Portal y se almacena de acuerdo a la estructura que éste maneja. En particular los campos número, título, descripción, ícono, imagen, idioma y texto adicional son campos predefinidos de cada contenedor. El texto libre puede formatearse a través de etiquetas HTML." [3]*

La GDR tiene como finalidad brindar un servicio que ofrezca información a la población lo más actualizada posible y presentada de forma amigable. Sin embargo, actualmente se utiliza un proceso de ingreso de la información que presenta limitantes para mantener la información actualizada. Además, las formas de acceso actuales podrían ser más diversas y las metodologías de búsqueda presentes no son del todo efectivas.

Observando la finalidad de la GDR, y analizando la forma en la cual se actualizan los datos y las opciones de búsquedas de recursos, surgen las siguientes limitantes de la versión actual:

- *Problemas de actualización de la información.*  
El mecanismo existente para el ingreso de la información, no permite que la información se actualice de forma ágil. Esto podría provocar que el usuario final acceda a información incorrecta.
- *Gestión de roles y usuarios*  
La versión actual de la GDR no brinda un mecanismo de manejo de usuarios con diferentes funciones. Esto no permite descentralizar la gestión de recursos, dificultando la gestión ágil de los mismos. Tampoco se brindan usuarios para el público general, que permita por ejemplo personalizar información de interés.
- *Búsqueda de recursos.*  
Actualmente la GDR ofrece una búsqueda por texto y categorías. Sin embargo, estos métodos resultan poco intuitivos y en ocasiones no funcionan de la forma esperada. Además, se notó que resulta difícil encontrar recursos cuando no se sabe sus nombres de forma exacta.
- *Accesibilidad del sistema*  
La interacción con el sistema implica tener acceso a una computadora, ya que se ofrece solamente una interfaz optimizada para PC. Sin embargo, existe una parte del público objetivo que potencialmente tiene solo acceso a otro tipo de dispositivos, como por ejemplo celulares o tabletas.

## 1.2 Objetivos

El objetivo general de este proyecto es analizar, evaluar y proponer alternativas tecnológicas para las limitantes actuales de la GDR.

Dentro de los objetivos específicos propuestos se encuentran:

- *Análisis detallado de la problemática actual de la GDR*  
Analizar las necesidades actuales de la GDR interactuando con el personal del MIDES.
- *Investigar tecnologías y soluciones similares*  
Investigar las tecnologías necesarias para la realización del proyecto. Identificar y analizar soluciones similares con el fin de obtener ideas que sean aplicables para la nueva versión.
- *Propuesta de alternativas tecnológicas*
  - Ofrecer un buscador completo y simple de usar, que funcione correctamente ante errores de escritura, mostrando los resultados que más se asemejen al texto escrito.
  - Proveer mecanismos que permitan mantener la GDR lo más actualizada posible. En particular, se planteó el objetivo de que el sistema detecte de forma automática cambios en páginas web externas que tengan información oficial de los recursos.
  - Proveer mecanismos que permitan manejar diferentes tipos de usuario que colaboren sistemáticamente a la actualización de la GDR. Estos pueden ser usuarios funcionarios del MIDES así como también usuarios responsables de los recursos específicos.
  - Proveer una interfaz web más amigable y que pueda ser accedida desde dispositivos móviles. De este modo se podrá llegar a un mayor volumen de personas.
- *Prototipación y validación*  
Realizar un prototipo de solución que integre y permita validar las propuestas de alternativas tecnológicas planteadas.

## 1.3 Aportes

Realizando una evaluación del producto y de las decisiones que se tomaron, podemos afirmar una serie de aportes que nuestro proyecto realizó:

- *Propuestas de mecanismos que permiten mantener la información de la GDR más actualizada*

- *Detección de cambios*

El sistema posee la facultad de detectar cambios en páginas web externas a la GDR. Esto permite que, en caso de existir cambios en la información extraída de páginas oficiales de los recursos, no queden inadvertidos. De este modo el usuario tiene otra alternativa para mantener la información de la GDR actualizada

- *Proceso de metodología de trabajo*

Se definió un proceso de actualización de la información de recursos el cual maneja diferentes tipos de usuarios y responsabilidades de los mismos. Se manejan usuarios que pueden manipular directamente los recursos, y otros habilitados a realizar propuestas de cambio que pueden ser validadas, totalmente, parcialmente o rechazadas.

- *Interfaz amigable y diferentes tipos de acceso*

- *Interfaz optimizada para dispositivos móviles.*

Se creó una parte de la interfaz de la GDR específica para dispositivos móviles, donde se detecta el ingreso de este tipo de dispositivos y se ofrece una interfaz optimizada para la búsqueda y visualización de recursos

- *Visualización de la ubicación de los recursos.*

Se utilizaron mapas en los cuales se pueden ubicar geográficamente los recursos.

- *Buscador*

Se realizó un buscador de texto con un algoritmo de distancia. Esto permite prevenir los errores de ingreso de caracteres, faltas ortográficas o escritura aproximada. Asimismo se introdujo la posibilidad de ingreso combinado de categorías. Las categorías están representadas en forma taxonómica.

- *Desarrollo de prototipo avanzado*

Se implementó un prototipo avanzado el cual integra y permite validar las propuestas tecnológicas. Esto brinda una base completa para la generación de la nueva GDR.

Asimismo se generó una máquina virtual con este producto en funcionamiento, la cual está a disposición para instalarse en cualquier servidor. Además se implantó el producto en servidores de la nube lo que permite el acceso a todo público.<sup>1</sup>

- *Relevamiento de problemáticas y tecnologías*

---

<sup>1</sup> <http://jbossas-gdrv2.rhcloud.com/GDR-web/>

En la primera parte del proyecto se hizo un relevamiento de requerimientos y tecnologías, que pueden ser útiles en proyectos similares. El análisis realizado puede servir como punto de partida para aquellos proyectos que tengan puntos de contacto con el proyecto desarrollado.

- *Componentes de software reutilizables*

El proyecto GDR internamente maneja el detector de cambios web, el cual está modularmente implementado. Realizando modificaciones menores, este componente podría dejarse disponible en una librería.

## 1.4 Organización del documento

En el capítulo 2 se presentan los conceptos y tecnologías utilizadas a lo largo del proyecto, con el objetivo de llevar al lector a entender sobre los temas de interés del proyecto GDR.

En el capítulo 3 se describen los requerimientos más destacados detectados en las reuniones de relevamiento con los funcionarios del MIDES (Cliente).

En el capítulo 4, se presenta la solución planteada, describiendo más en detalle qué solución finalmente se tomó para cada uno de los problemas. En dicho capítulo se entra más en detalle de cómo fue la implementación de ciertas partes de la GDR.

En el capítulo 5, se muestra cómo fue el proceso de Gestión del proyecto, dando detalles de cómo se organizó, cómo evolucionó y también se explican las técnicas de gestión utilizadas.

En el capítulo 6, se muestra cómo se hizo una validación de la interfaz web con el cliente.

Finalmente en el capítulo 7 se encuentran las conclusiones y trabajos a futuro que dejó el proyecto GDR.



## 2 MARCO CONCEPTUAL

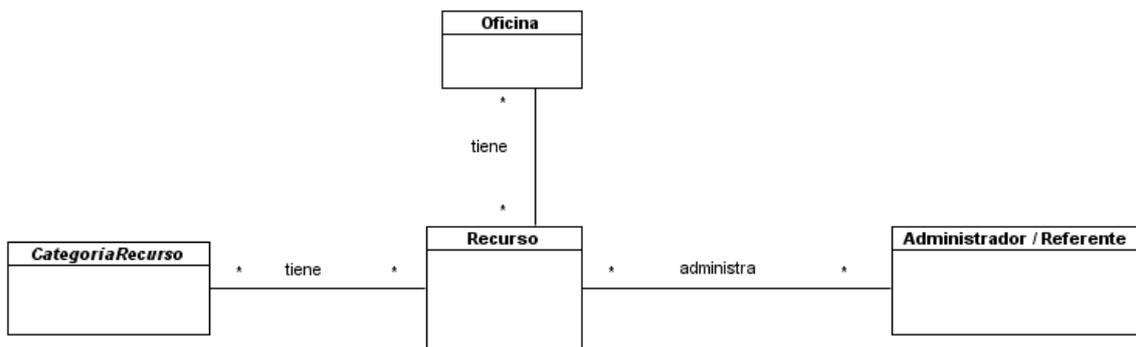
En este capítulo se enumeraran los conocimientos necesarios que se tuvieron que adquirir para llegar a la solución planteada. Ya sean conocimientos técnicos como conocimientos de funcionamiento de la GDR.

### 2.1 Guía de Recursos del MIDES

*“La Guía de Recursos Sociales es una herramienta activa de difusión de información sistematizada sobre servicios, programas, centros de atención e instituciones, ya sean gubernamentales o no gubernamentales, que focalizan su labor en la mejora de las condiciones de vida de nuestra población. Su objetivo principal es fortalecer la democratización de la información vinculada al cumplimiento de los principales derechos de todos los ciudadanos y a mejorar la calidad y la frecuencia de la actualización de la misma” [1]*

La guía se puede acceder a través de Internet y está dirigida al público en general (beneficiarios de los recursos, asistentes sociales, etc). Por este motivo, la guía brinda información detallada que incluye, por ejemplo, direcciones y horarios de atención de las oficinas donde se puede acceder a los recursos.

La Figura 2.1 muestra el modelo conceptual de la realidad actual



**Figura 2.1 – Modelo conceptual**

A continuación se detallan los conceptos más importantes:

- **Recurso:** es la unidad de información central de la GDR y representa lo que sería un servicio, institución, programa, etc.
- **Oficina:** representa un lugar físico de un recurso con ubicación propia. Un recurso puede no tener o tener muchas oficinas en el Uruguay.
- **Categoría:** es la clasificación taxonómica para un conjunto de recursos. Un recurso puede pertenecer a cualquier cantidad de categorías. Esto puede ser útil para hacer una rápida referencia a un conjunto de recursos y con ello buscar un recurso específico, como así también describir a determinado recurso.
- **Administrador:** es el tipo de usuario que se encarga de gestionar directamente la información de los recursos, toda la información editada en la GDR antes será validada por algún usuario de este tipo. En la práctica serán empleados del área social del MIDES.
- **Referente:** es el tipo de usuario que se encarga de gestionar de forma indirecta la información de los recursos, realizando propuestas de recursos y oficinas para que algún administrador las verifique y las publique o no definitivamente. En la práctica serán personas que conocen mejor un recurso específico o un conjunto de recursos específicos, por ej: trabajadores directos de los recursos los cuales son de confianza para editar determinados recursos.

## 2.2 Tecnologías para desarrollo empresarial

Esta sección describe las tecnologías para el desarrollo empresarial.

### 2.2.1 Plataformas de desarrollo empresarial

Si hablamos de software empresarial, hablamos de un software que es desarrollado para el uso de una empresa específica, y no es un software genérico que intenta venderse a múltiples usuarios individuales. No existe una única característica del software empresarial, pero por lo general son accesibles desde una red internet o intranet, ofrecen robustez, escalabilidad y servicios de comunicación con otros sistemas. [4] [5]

Con el pasar de los años han existido diversas plataformas como por ejemplo CORBA, pero hoy en día las más populares e importantes son Microsoft .Net y JavaEE. [6]

#### Java Platform Enterprise Edition (Java EE)

JavaEE es un conjunto de especificaciones estándar que describen componentes de aplicación, APIs, contenedores de ejecución y servidores de aplicación. [7]

La Figura 2-2 muestra la arquitectura de la 6ta versión de JavaEE, la cual brinda un conjunto de herramientas con la que da soporte a la construcción de aplicaciones.

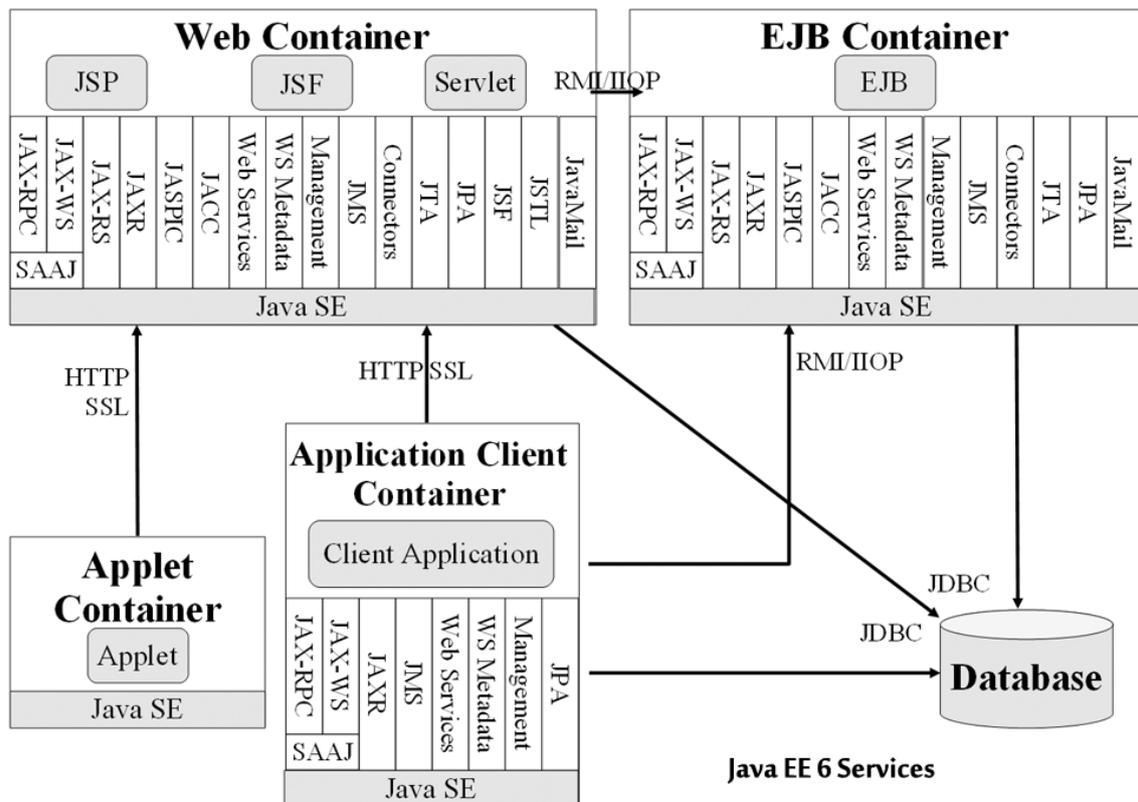


Figura 2-2 - Arquitectura JavaEE 6 [8]

La Figura 2.3 muestra las diferencias que existen entre las distintas versiones de JavaEE, su evolución y posibilidades para desarrollar aplicaciones sobre dicha plataforma.

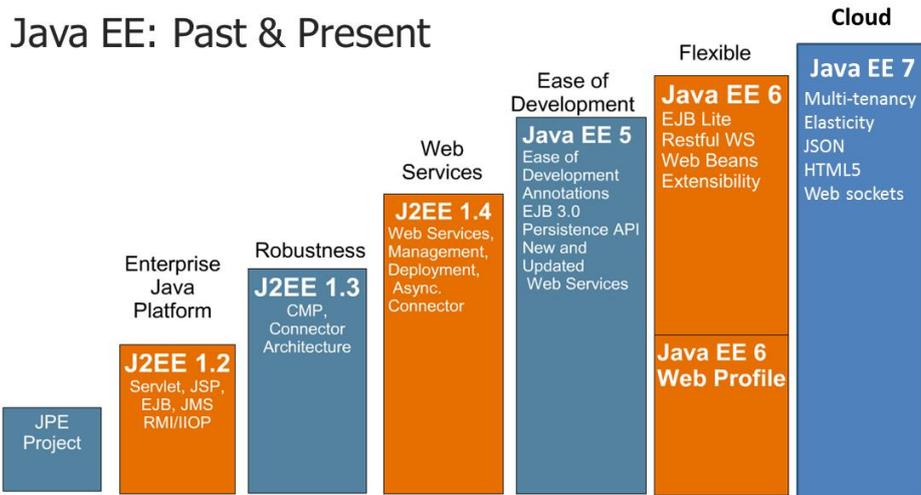


Figura 2.3 - Evolución JavaEE [57]

Las herramientas que ofrece la plataforma se ejecutan en un servidor de aplicaciones, estos servidores implementan las especificaciones de Java EE.

### Servidores de aplicación

Un servidor de aplicación es una tecnología básica que proporciona la infraestructura y servicios clave a las aplicaciones alojadas en un sistema. Entre los servicios habituales de un servidor de aplicaciones se incluyen los siguientes [9]:

- Agrupación de recursos (por ejemplo, agrupación de conexiones de base de datos y agrupación de objetos)
- Administración de transacciones distribuida
- Comunicación asincrónica de programa, normalmente a través de colas de mensajes
- Un modelo de activación de objetos oportuno
- Interfaces de servicios Web XML automáticas para tener acceso a objetos de empresa
- Servicios de detección de errores y estado de las aplicaciones
- Seguridad integrada

Una aplicación desarrollada con las tecnologías JavaEE permite ser desplegada en cualquier servidor de aplicaciones o servidor web que cumpla con el estándar. Existen diversos servidores de aplicación entre ellos: JBoss, Glassfish, WebSphere.

### Java Server Faces (JSF)

Dentro de las herramientas que ofrece la plataforma JavaEE existe JSF, el cual es un framework estándar de componentes para la construcción de interfaces de usuario en aplicaciones Java. Desarrollado por la comunidad de Java bajo la solicitud JSR – 314, JSF tiene aportes del grupo de expertos. JSF está diseñado de forma que las herramientas puedan impulsar a que el desarrollo web sea aún más fácil. JSF ejecuta en el servidor y es desplegado en lado del cliente. [10]

JSF simplifica el desarrollo de páginas web, “La tecnología JavaServer Faces nos permite construir aplicaciones Web que implementan una separación entre el comportamiento y la presentación tradicionalmente ofrecidas por arquitectura UI del lado del cliente.” [11]

Ofrece elementos gráficos que se auto generan en HTML puro, validadores y conversores de datos. Para la administración del estado del lado del servidor, brinda los llamados Managed Beans.

### PrimeFaces Framework (PF)

Primefaces es un framework para JSF que ayuda a realizar páginas web JSF de una forma más ágil aún. PF es una librería liviana y sin dependencias de otras librerías del lado del servidor, y del lado del cliente utiliza la librería JavaScript JQuery. A su vez intenta ser lo más performante posible, brinda componentes avanzados en complejidad pero muy simples de usar. Hoy en día es el más usado en comparación con sus competidores según google trends [12], [13].

A modo de ejemplo, en lo que a mapas se refiere, ofrece el componente específico de llamado `<p:gmap>` [14]. Este componente está basado en google maps, aunque no ofrece la totalidad de las funcionalidades de google maps, pero siempre es posible utilizar su API nativa para cualquier detalle específico del sistema.

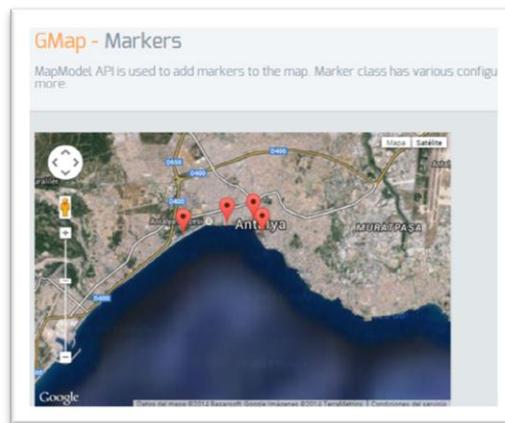


Figura2.4 – Ejemplo de uso de `<p:gmap>` extraído de demo de PF

### Persistencia

En general una aplicación empresarial se apoya en una base de datos relacional. Entre los manejadores de base de datos open source más utilizados se encuentra: MySQL y PostgreSQL.

#### MySQL

MySQL es un sistema de gestión de base de datos relacional con todas las funciones (RDBMS) que compite con los gustos de Oracle DB y Microsoft SQL Server. MySQL es patrocinado por la compañía sueca MySQL AB, que es propiedad de Oracle Corp. Sin embargo, el código fuente de MySQL está disponible gratuitamente porque fue desarrollado originalmente como freeware. MySQL está escrito en C y C++ y es compatible con los principales sistemas operativos.

#### PostgreSQL

PostgreSQL es un sistema de código abierto objeto-relacional de gestión de base de datos (ORDBMS) que no es propiedad o está controlado por una empresa o individuo. Dado que el software PostgreSQL es de código abierto, el mismo fue desarrollado principalmente a través de un esfuerzo coordinado en línea por una comunidad global activa de desarrolladores, aficionados y otros voluntarios.

Lanzado por primera vez a mediados de la década de 1990, PostgreSQL está escrito en C. Sus principales competidores incluyen Oracle DB, SQL Server y MySQL.

## 2.2.2 Tecnologías para Información Geográfica

### Información Geográfica

*“Un mapa es una representación a escala de una selección de entidades sobre la superficie de la Tierra. La noción de “mapa” se utiliza en Matemática y Computación como una transferencia de información de una forma en otra, de la misma forma que un cartógrafo transfiere información desde la superficie de la Tierra a un papel.” [15]*

La información geográfica refiere datos georreferenciados. La georreferenciación es el proceso de asignarle a un dato coordenadas sobre la tierra. Como la forma de la tierra es geoide, se usa un sistema de coordenadas basado en grados considerando latitud y longitud.

Existen varios modelos de datos geográficos, entre ellos los raster, triangulares y vectoriales, cada uno de ellos es más adecuado dependiendo el tipo de dato a trabajar.

**Datos Vectoriales:** *“Representación geométrica de características geográficas en forma precisa y compacta, utilizando puntos, líneas y polígonos.” [15]* Las coordenadas conforman todos los tipos de datos vectoriales.

**Sistemas de Información Geográfica (SIG):** *“Un Sistema de Información Geográfica es la combinación de cinco componentes: personas especializadas, datos descriptivos y espaciales, métodos analíticos, hardware y software; organizados para analizar, manipular, procesar, almacenar, generar y visualizar todo tipo de información referenciada geográficamente.” [16]*

Actualmente los SIG son sistemas distribuidos que están disponibles en internet. Para ello son necesarios estándares, protocolos, servidores de mapas y también las bases de datos espaciales.

### Servidores de mapas

Los servidores de mapas son componentes de software que implementan los estándares y protocolos necesarios. Generalmente los servidores de mapas se ejecutan dentro de servidores web o servidores de aplicaciones: MapServer, GeoServer, ArcGISServer. Otros servidores de mapas como Google Maps manejan otra arquitectura, esta arquitectura consiste en consultar la mayoría de los datos geográficos directamente a los servidores de google.

La visualización de mapas de Google Maps es posible por medio de los navegadores web, los cuales consumen un API Javascript.

### Bases de datos Espaciales

Las bases de datos espaciales son Bases de Datos que manejan objetos geográficos. Permiten guardar dichos datos, así como también manipularlos. Un ejemplo de base de datos espacial es PostgreSQL con su extensión PostGIS.

En pocas palabras, PostGIS añade funciones espaciales tales como la distancia, área, unión, intersección, y además tipos de datos espaciales a la base de datos PostgreSQL.

### 2.2.3 Web Services

Los Web Services son un conjunto de protocolos y estándares que permiten la comunicación entre diferentes aplicaciones. La principal característica de los Web Services es que las aplicaciones que se comunican no deben ser necesariamente desarrolladas en el mismo lenguaje de programación.

#### Web Service - Simple Object Access Protocol (SOAP)

El Web Services Description Language (WSDL) contiene y describe el conjunto de normas comunes para definir los mensajes, enlaces, operaciones y ubicación del Web Service. WSDL es un tipo de contrato formal para definir la interfaz que ofrece el Web Service.

Web Service SOAP son útiles en el manejo de procesamiento asincrónico e invocación. SOAP soporta varios protocolos y tecnologías, incluyendo WSDL, XSD, SOAP, WS-Addressing. Los datos se pueden obtener en diversos formatos como JSON, XML, Base64, etc. [17] [18] [19]

#### Web Service - Representational State Transfer (REST)

Los Web Services REST están contruidos para funcionar mejor en la Web. En el estilo de arquitectura REST, datos y funcionalidad se consideran los recursos y se accede mediante identificadores uniformes de recursos (URI), típicamente enlaces en la Web. Los recursos son accionados mediante el uso de un conjunto de operaciones simples y bien definidas. El estilo arquitectónico REST se restringe a una arquitectura cliente / servidor y está diseñado para utilizar un protocolo de comunicación sin estado, normalmente HTTP. En el estilo de la arquitectura REST, los clientes y servidores intercambian recursos mediante el uso de una interfaz y protocolo estandarizados. [20]

##### *Características de los Web Services REST:*

- Son completamente sin estado.
- No es un estándar, ya que es un estilo arquitectónico, pero de igual manera esta basado en estándares como: HTTP, URL, Representacion de Recursos: XML, HTML, GIF, JPG, JPEG, JSON, tipos MIME: text/html, text/xml.
- Al igual que HTTP funcionan con los pedidos de PUT, GET, DELETE, POST y HEAD.
- El productor y el consumidor de servicios necesitan tener una comprensión común del contexto, ya que no hay un conjunto estándar de reglas para describir la interfaz de estos WebServices.

##### *Ventajas de Web Services REST:*

- REST es útil para dispositivos tales como PDA y móviles para los que sobrecarga de parámetros adicionales como los encabezados y otros elementos de SOAP son menores. O sea las respuestas contienen solo la información que se necesita.
- Servicios REST son fáciles de integrar con los sitios web existentes y están expuestos con XML para que las páginas HTML pueden consumir la misma con facilidad. Casi no hay necesidad de modificar la arquitectura web existente.
- Proporcionan una buena infraestructura de almacenamiento en caché a través de HTTP GET método (para la mayoría de los servidores). Esto puede mejorar el rendimiento, si los datos devuelve el servicio Web no se altera con frecuencia y no de naturaleza dinámica.

**Desventajas de Web Services REST:**

- No hay un estándar en sus respuestas por lo que no se definen tipos de datos.
- No provee seguridad, en caso de que se quiera incorporar seguridad en la utilización de este servicio, se debe generar un canal HTTPS para la comunicación.

## 2.3 Desarrollo Dispositivos Móviles

Los dispositivos móviles que existen hoy en día, poseen un sistema operativo el cual permite el desarrollo de aplicaciones nativas. Estas aplicaciones son desarrolladas en diferentes lenguajes de programación para cada sistema operativo.

### 2.3.1 Aplicaciones para dispositivos móviles

Se describen los distintos sistemas operativos y lenguajes de programación que manejan los dispositivos móviles que hay hoy en día.

#### Android OS

Android es un sistema operativo de código abierto que se utiliza principalmente en dispositivos móviles. Esta implementado principalmente en Java y basado en el sistema operativo Linux. [21]

Esto permite realizar un desarrollo para aplicaciones móviles que utilicen este sistema operativo, con prestaciones como la geolocalización propia del dispositivo móvil. Además posee una pequeña base de datos, brindando la posibilidad de un funcionamiento offline.

#### IOS

El iOS es el sistema operativo diseñado por Apple para sus productos, iPhone, iPad, iPod Touch, y Apple TV. Otros dispositivos como el iPod Nano y el iWatch utilizan otro sistema más básico y dirigido a una función más específica basado en iOS porque incorpora algunos de sus gestos e iconos y además se pueden sincronizar con teléfonos o Tablets. [22]

Los lenguajes de programación que se pueden utilizar para desarrollar sobre este sistema operativo son: Objective-C y Swift.

Objective-C es el lenguaje de programación principal que se utiliza cuando la escritura de software para OS X y iOS. Es un superconjunto del lenguaje de programación C y proporciona capacidades orientadas a objetos y un tiempo de ejecución dinámico. [23]

Swift es un lenguaje de programación multiparadigma creado por Apple enfocado en el desarrollo de aplicaciones para iOS y Mac OS X.

#### Windows Phone

Para el desarrollo de aplicaciones sobre Windows Phone contamos con el lenguaje de programación C#, el cual se asemeja a lo que es el lenguaje de programación Java en términos generales.

Se provee una SDK (Software development kit) la cual ofrece una API para poder desarrollar contenedores y controles para aplicaciones móviles.

### 2.3.2 Interfaz web para dispositivos móviles

Con el aumento del uso de los smartphones, crear un sitio web optimizado para los usuarios de estos dispositivos es fundamental en la gestión de un sitio web.

Existen dos principales enfoques para el desarrollo de interfaces web mobile [24]:

- **Diseño receptivo:** El sistema detecta la resolución que tiene la pantalla del dispositivo que se está accediendo y este reajusta la pantalla para el correcto despliegue del sitio.
- **Páginas separadas:** El sistema detecta el dispositivo que está accediendo y redirige la pantalla que más se ajusta para él.

Hoy en día existen varias opciones a la hora de desarrollar una página web adaptada para celulares y tablets. Entre ellas existe PrimeFaces Mobile (PFM) y JQuery mobile.

PFM sigue el diseño optimizado en **páginas separadas**, lo cual se sitúa mejor para aplicaciones JSF. PFM está basado principalmente en jQuery mobile y HTML5, pero a eso le suma las características de JSF, lo cual facilita la implementación.

Si ya se tiene experiencia en el desarrollo con componentes de PrimeFaces, esa experiencia es útil ya que la versión mobile de PrimeFaces dibuja de distinta forma (distinto código HTML final) los mismos componentes: datalist, menu, panel, accordion, tabview, dialog, growl, forms y más. Con lo que en definitiva un mismo código JSF generaría una interfaz optimizada para mobile. [25]

La Figura 2.5 muestra cómo se desplegarían en los diferentes dispositivos un sitio web. De esta manera se muestra cómo adaptar la misma página para diferentes dimensiones de pantalla y formas de interacción.

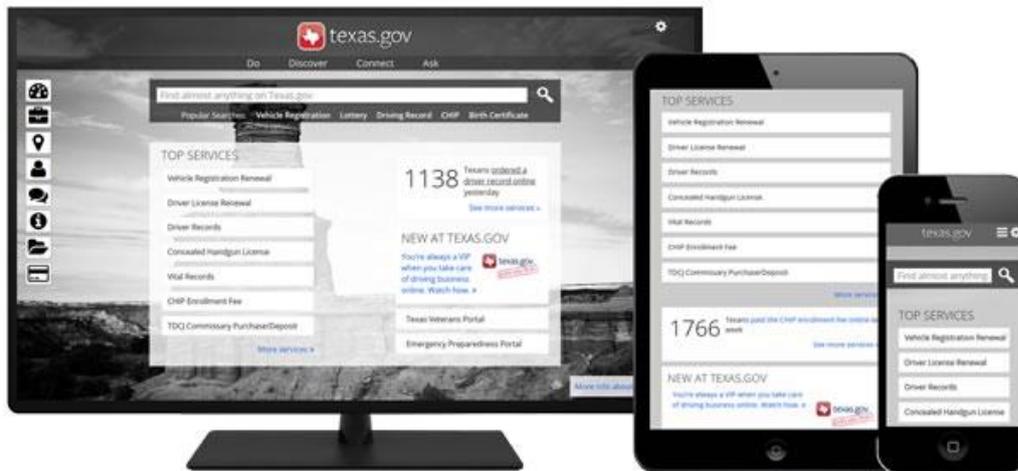


Figura 2.5 - Sitios Web en diferentes dispositivos

## 2.4 Conceptos de Seguridad

### Autenticación

Autenticación es el proceso que efectúa una entidad para probar su identidad

- Generalmente para autenticar se usa el par nombre de usuario y contraseña, pero existen otros métodos como: tarjetas, reconocimiento de voz, lectura de huellas digitales, etc.
- La autenticación no determina qué tareas el usuario puede hacer, la autenticación solamente identifica y verifica que persona o sistema es. [26]

### Autorización

Autorización es el proceso de determinar si una entidad previamente autenticada tiene permisos para realizar una acción o acceder a un recurso.

- Generalmente la autorización viene de la mano con la autenticación, el servidor debe tener idea qué cliente está solicitando acceso.
- En algunos casos no hay autorización, por ejemplo cuando todos los usuarios tienen acceso a todo, y también puede suceder que no se requiera autenticación ni autorización, por ejemplo esto sucede con la mayoría de las páginas web. [26]

### Resource Based Access Control (The new RBAC)

El control de acceso basado en roles, funciona accediendo directamente a los roles, esto es denominado acceso implícito. Es decir, cada vez que se quiere consultar un permiso para realizar determinada acción, se pregunta si el usuario tiene el o los roles permitidos.

Por otro lado “The new RBAC” propone una mejora, la cual consiste en acceder a los recursos (elementos del sistema) y no directamente a los roles. Esto es, preguntar si el usuario tiene permisos para acceder al recurso. Anteriormente se deben haber definido el conjunto de permisos que cada rol tiene. La gran ventaja es que si los permisos de los roles cambian, el código fuente no se ve afectado, cambiar los permisos se resume a un simple trabajo de configuración del sistema. [27]

Apache Shiro es un ejemplo de implementación de este tipo de acceso, pero a su vez Apache Shiro no exige el uso de este paradigma, ofrece ambas opciones anteriormente descritas.

### Ataques para descubrir contraseñas

Existen diversos métodos de ataque usados por los hackers a la hora de descubrir una contraseña, algunos de estos son:

- Dictionary
- Bruteforce
- LookupTables
- Reverse LookupTables
- RainbowTables

Se debe tratar de minimizar la posibilidad del alcance de dichos ataques y en el caso de ser posibles el tiempo computaciones sea muy grande. [28]

## Criptografía - Algoritmos de Hashing

Típicamente las contraseñas son persistidas en un archivo o en una base en datos con un algoritmo unidireccional MD5, SHA1, SHA256, SHA512. Esto se hace para que un hacker o un sysadmin no tengan acceso a las contraseñas, al menos no fácilmente.

## Criptografía - Agregando Salt

El salt es algo necesario hoy en día, se trata de una tira aleatoria de bytes del mismo tamaño que la salida del algoritmo de hash.

Esto sirve para hacer contraseñas débiles, menos débiles. De todos modos el sysadmin tiene acceso a los salts, por lo tanto puede intentar un ataque de fuerza bruta, pero de seguro es más difícil.

Es importante que sea aleatoria para evitar hashes iguales en la misma base de datos también

## Criptografía - Enlenteciendo ataques de fuerza bruta

### *keystrechting*

Además de lo anterior para hacer las contraseñas aún más seguras se puede usar algoritmos que enlentecan los intentos de ataques de fuerza bruta suficientemente, pero que el uso al momento del login sea rápido al ojo de un usuario. Incluso el keystrechting hace que los GPUs sean lentos al hacer ataques de fuerza bruta [29]. PBKDF2 y BCrypt son ejemplos de keystrechting.

Si se usan estos algoritmos en aplicaciones web, se debe tener precaución. Un atacante podría generar un Denial of Service, esto es debido al costo de procesamiento que implica. Un captcha en la página de autenticación podría ser una solución posible, pero no una experiencia placentera para el usuario. Otra solución podría ser realizar el algoritmo en el lado del cliente con JavaAcript.

### *Múltiples iteraciones*

También se pueden enlentecer haciendo múltiples iteraciones de hashing como hash(hash(hash(...))), aunque dicen que muchos algoritmos como SHA256 son fácilmente paralelizables con GPUs, lo que hace que esto de las iteraciones no sea algo tan efectivo, por eso pareciera ser mejor el keystrechting.

## Cryptographically Secure Pseudo-Random Number Generator (CSPRNG)

Es un generador pseudorandómico seguro de números. Con esto se pueden generar claves y salts de forma aleatoriamente seguras, usando una implementación de CSPRNG nos aseguramos de tener un generador realmente impredecible, muchos lenguajes de programación proveen de un CSPRNG, java tiene el SecureRandom [30].

## Apache Shiro

Apache shiro es un framework de seguridad, poderoso y fácil de usar que ofrece Autenticación, Autorización, Criptografía y Manejo de Sesiones. En sus inicios este framework nació a base de carencias que se le encontraron a JAAS (*Java Authentication and Authorization Service*), como por ejemplo la dificultad de utilización, pero además encontraron que no brindaba soporte al cambio constante de los permisos sobre los roles de usuarios. [31] [32]

### VtPassword

Es una librería de java que permite el chequeo de passwords basado en reglas, y también permite la generación aleatoria de passwords basado en estas reglas, a continuación se dan las siguientes reglas de ejemplo [33]:

- Debe tener determinada cantidad de caracteres numéricos
- Debe tener determinada cantidad de letras minúsculas
- Debe tener determinada cantidad de letras mayúsculas
- Debe tener determinada cantidad de caracteres no alfanuméricos

Esta librería puede ayudar a generar passwords más seguros que una simple palabra conocida, la cual seguramente su hash esté en las RainbowTables.

## 3 ANÁLISIS

En este capítulo se detallan algunos de los requerimientos que fueron relevados a través de varias reuniones con el cliente.

### 3.1 Requerimientos

Se presentan los requerimientos relevados y el modelo de dominio resultante.

#### 3.1.1 Relevamiento de Requerimientos

##### *Gestión de Usuarios y Control de acceso*

Se requiere un manejo de usuarios en el cual se pueda definir qué funcionalidades son accesibles a qué usuario.

- **Usuario Común:** usuario que no se loguea.
- **Usuario Logueado:** usuario que se loguea (con o sin facebook) y puede asociarse a un conjunto de recursos los cuales le interesa.
- **Referente de Campo:** Usuario que se loguea y puede ingresar nuevos recursos y tiene permisos sobre determinados recursos los cuales puede modificar. Todas las modificaciones que estos usuarios hagan deben ser validadas por otro usuario que tenga rol de administrador antes de ser publicadas en la web.
- **Administrador:** Usuario que tiene permisos de modificación sobre la totalidad de los recursos y también puede crear nuevos, además puede aumentar los permisos sobre otros usuarios.
- **SuperAdministrador:** Usuario que tiene permisos para crear y borrar Administradores.

En primera instancia existirá al menos un SuperAdministrador que tendrá permisos para crear administradores.

Los principales protagonistas en la GDR serán los administradores y referentes quienes serán los usuarios que trabajarán en conjunto para mantener la información actualizada. También servirá de ayuda la Información Externa y las notificaciones automáticas de cambios.

A continuación se detallan las funcionalidades de cada usuario:

Administradores:

- Alta/Baja/Modificación de Usuario Logueados y Referentes de Campo
- Modificar su perfil (recursos que monitorea) y sus datos.
- Alta/Baja/Modificación recursos.
- Validar/Rechazar/Consultar propuestas de recursos.
- Configurar frecuencia de revisión de cambios en página de recurso.
- Ver muro de la GDR (notificaciones de cambios y propuestas de recursos pendientes que monitorea)
- Alta/Baja/Modificación de Categorías
- Alta/Baja/Modificación Información Externa.

Referentes:

- Modificar su perfil (recursos que monitorea) y sus datos.
- Alta/Baja/Modificación de propuestas de recursos.
- Ver la página principal de la GDR junto con las notificaciones de cambios

Usuario Logueado:

- Consultar/Buscar recursos
- Ver la página principal de la GDR junto con los últimos recursos consultados y recursos de interés
- Modificar su perfil (recursos de interés) y sus datos

Usuario Común (Sin registro en el sistema):

- Darse de alta a sí mismos y obtener un Usuario Logueado, dicho de otro modo registrarse en el sitio web de la GDR
- Consultar/Buscar recursos

### *Gestión de Recursos*

Se necesita incorporar herramientas para poder crear, modificar y eliminar un recurso en el sistema.

Si el recurso tiene más de un lugar físico, éste puede ser representado además con un conjunto de oficinas. Las oficinas representarán las diferentes ubicaciones en el territorio que un recurso puede tener, junto con sus datos específicos que podrían diferir como por ejemplo un “horario de atención”.

Para ayudar a las búsquedas, la configuración de notificaciones y configuración de perfiles, también un recurso tendrá un conjunto de categorías.

Es necesario que se permita una búsqueda eficiente para todo tipo de usuarios sobre los recursos, además de que los recursos mantengan visible la cantidad de visitas que tuvo en general.

Para los usuarios logueados, el sistema deberá desplegar los últimos recursos visitados por él.

### *Gestión de Información Externa*

Algunos recursos poseen sitios externos al MIDES donde se extrae información que se incluye en la GDR. Se requiere se notifique a los usuarios que monitorean un recurso, si el sitio externo sufrió un cambio o no de la información que expone, y así de esta manera brindar un mecanismo para que la GDR cuente con la información actualizada.

### *Gestión de Notificaciones Automáticas*

Se necesita instalar un sistema de notificaciones automáticas las cuales notifiquen a los usuarios correspondientes en los siguientes casos:

- Notifique a los monitores sobre cambios en los sitios asociados a los recursos que la GDR tiene información.
- Notifique a los monitores si un recurso no se actualizó por un tiempo determinado (configurable).
- Notificar a los administradores cuando se genera una propuesta de cambio de un recurso.

### *Gestión de Categorías*

Cada recurso se encuentra dentro de una o más categorías de cualquier tipo. Se necesita una representación jerárquica de categorías para poder agruparlos de forma comprensible y describirlos. La representación de las categorías debe ser jerárquica para permitir nombrar de forma general un conjunto de categorías. Por ejemplo, la categoría “consumo” si se quiere puede ser sub-categoría de “salud”, y “Bella Unión” si se quiere puede ser subcategoría de “Artigas”.

Existen tres tipos de categorías:

- **Área**  
Es una categorización conceptual que se hace sobre los recursos. Los recursos se agrupan según su similitud.  
Ejemplos: salud, vivienda, trabajo.
- **Territorio**  
Es una categorización de ubicaciones geográficas. Los recursos se agrupan por cercanía geográfica.  
Ejemplos: departamentos, barrios, zonas específicas.
- **Población Objetivo**  
Es una categorización de personas. Los recursos se agrupan por el tipo de persona que está destinado.  
Ejemplos: jóvenes, adultos, adultos mayores, hombres, mujeres.

La principal utilidad de las categorías es que posibilitan otra forma de búsqueda además de la búsqueda por ingreso textual, como por ejemplo: “Recursos de la población objetivo jóvenes, del área salud, y del territorio barrio cordón”

Además de facilitar la búsqueda de recursos permite a los administradores y referentes monitorear. Los usuarios logueados pueden interesarse de forma general:

- “un administrador monitorea los recursos de Maldonado”
- “un referente monitorea los recursos del área vivienda”



## 3.2 Sistemas informáticos con objetivos similares

Como parte del análisis de este proyecto, se decidió investigar sitios web que ofrecieran servicios que tengan un objetivo similar al objetivo que tienen la GDR. Los países de los sistemas analizados son: Brasil, Chile, Argentina y Estados Unidos.

### 3.2.1 Sistema de Brasil

#### Descripción

“Relatórios de Informações Sociais” es el nombre del sistema de Brasil. Este sistema brinda información estadística de los estados.

Dado un estado algunos ejemplos de la información que ofrece son:

- Área total del territorio
- Índice de pobreza
- Presupuestos de asistencia social
- Cantidad de familias que poseen seguridad alimentaria
- Índices de frecuencia de actualización catastral

El sistema que ofrece Brasil tiene una finalidad estadística y cuenta con dicha información separada geográficamente entre los diferentes Estados, Ciudades, Municipios. No se proporciona una guía de organizaciones y programas sociales como si lo ofrece la GDR.

La Figura 3.2 muestra la página principal del sitio que posee Brasil cuyo fin es el de brindar información a la población sobre los estados del país. [34]



Figura 3.2 - Página Inicial del sitio de Brasil

Seleccionando la opción Programas de informes y acciones MDS que se encuentra en el menú “Relatórios de Informações” ubicado a la izquierda de la pantalla de la figura 3-2, redirige a la pantalla específica como se muestra en la figura 3.3:

**Relatório de Programas e Ações MDS**

Para obter dados dos estados clique sobre uma UF e depois sobre o botão processar. Caso seja necessário gerar um município selecione-o na listagem do estado.

Brasil AC AL AM AP BA CE DF ES GO MA MG MS MT  
PA PB PE PI PR RJ RN RO RR RS SC SE SP TO

Estado selecionado: **MARANHÃO**

Selecione o Município

ESTADO DO MARANHÃO  
AÇAILÂNDIA  
AFONSO CUNHA  
ÁGUA DOCE DO MARANHÃO  
ALCÂNTARA

Ano: 2014

Código

**CONSULTAR**

Figura 3.3 - Página de Programas e Informes MDS

En la pantalla se encuentra en el lado izquierdo el mapa de Brasil delimitado por los estados correspondientes. En el lado derecho figura una botonera, la cual representa los estados en el mapa y permite seleccionar los Estados de Brasil. Si se selecciona uno de los Estados, despliega su conjunto de municipios en el listado que se visualiza por debajo de la botonera.

Se selecciona el municipio de “Maranhão” y se llena los campos de control para su consulta, entonces el sistema redirige a la página que se muestra en la Figura 3.4

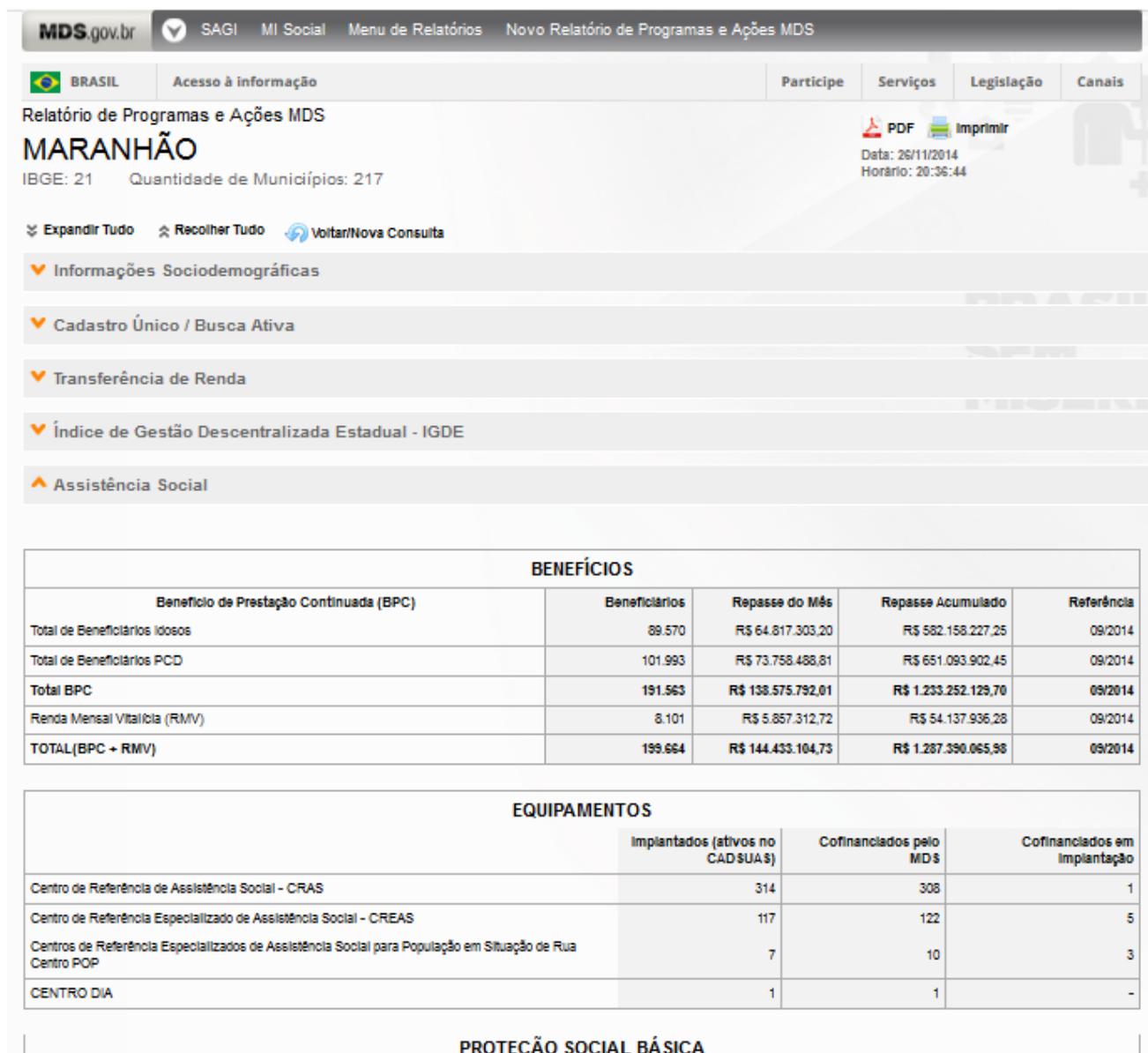


Figura 3.4 – Información sobre el Municipio

Aquí se puede ver información relevante sobre el municipio seleccionado. Está orientado a dar información del tipo estadístico.

### 3.2.2 Sistema de Chile

#### Descripción

El sitio de Chile es un sistema con una finalidad similar a la que tiene la GDR. En él se encuentra información sobre programas sociales existentes en Chile. [35]

La Figura 3.5 muestra cómo es la página principal del sitio:

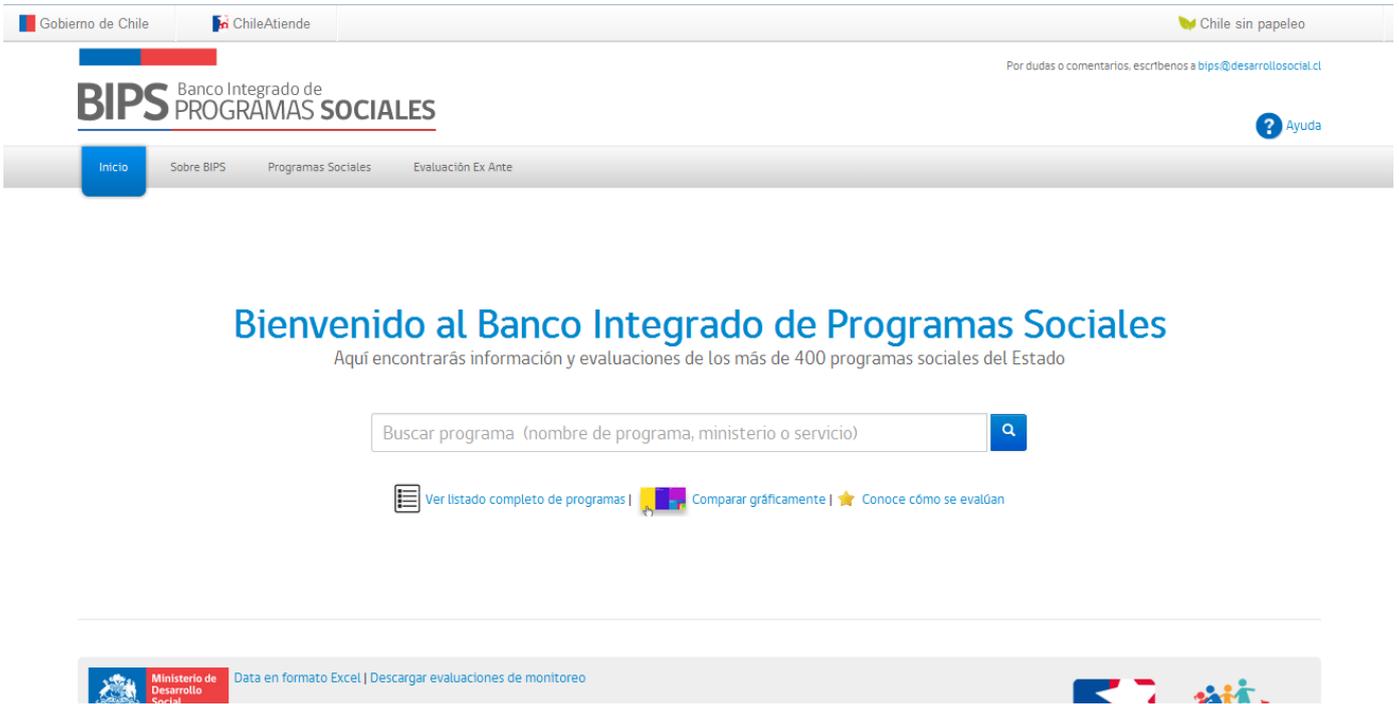


Figura 3.5 - Página Inicial Chile

La página principal brinda un buscador el cual dado el diseño de la página, es la funcionalidad principal que ofrece este sitio. En el buscador se realiza una búsqueda por “Deporte” obteniendo el resultado que muestra Figura 3.6:

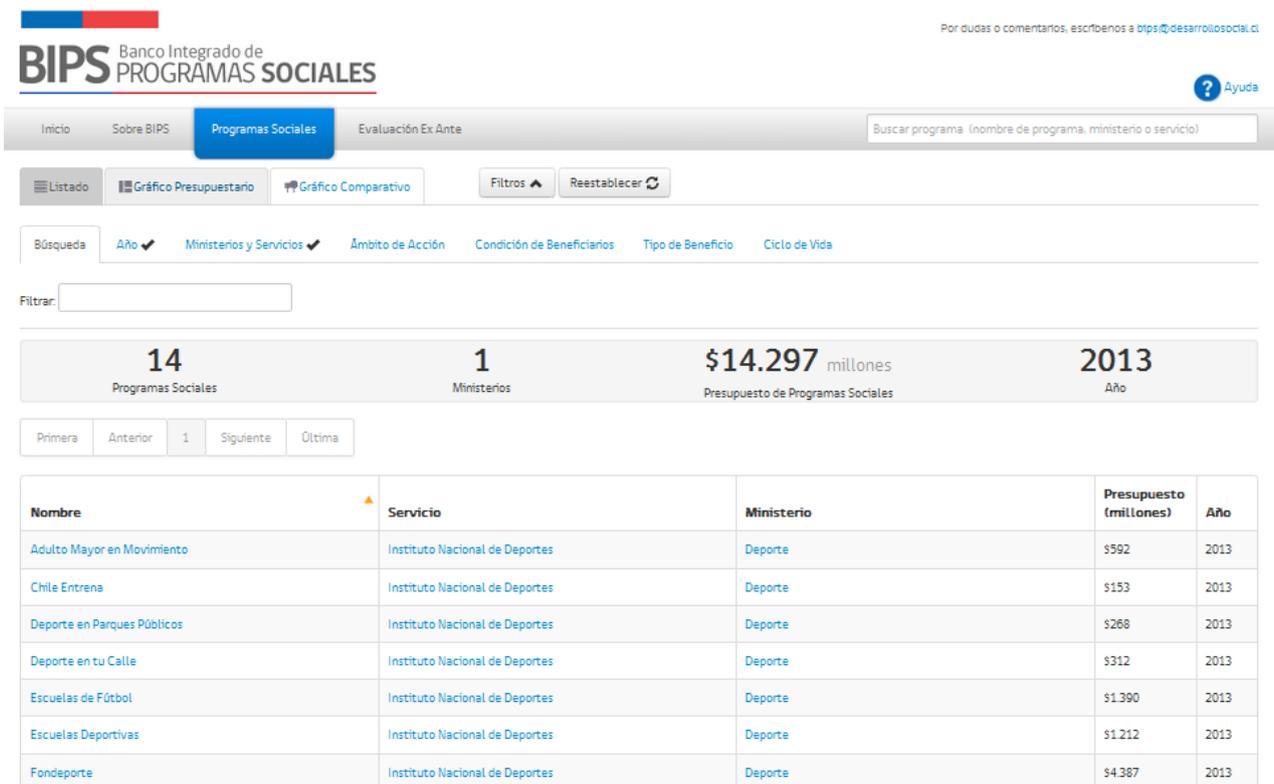


Figura 3.6 - Búsqueda de “deporte”

La página cuenta con un filtro ubicado en la izquierda, el cual permite filtrar los programas sociales listados. Debajo de este filtro se encuentra un listado de los programas sociales resultantes de la búsqueda realizada, pudiendo ingresar al detalle de cada uno de los programas sociales. Si se selecciona “Adulto Mayor en Movimiento”, el sitio redirige a la página con el detalle de la información que se tiene sobre ese programa social como se muestra en la Figura 3.7:

**BIPS** Banco Integrado de PROGRAMAS SOCIALES

Por dudas o comentarios, escribenos a [bips@desarrollosocial.cl](mailto:bips@desarrollosocial.cl)

Inicio Sobre BIPS **Programas Sociales** Evaluación Ex Ante

Inicio / Programas Sociales / Deporte / Instituto Nacional de Deportes / Adulto Mayor en Movimiento

## Adulto Mayor en Movimiento

**Ministerio** Deporte [Descargar Ficha 2013](#)

**Servicio** Instituto Nacional de Deportes

**Tipo Programa** Vigente

**Año** 2013

**Página Web** [www.IND.cl](http://www.IND.cl)

### Descripción y Objetivos del Programa

El programa busca que los adultos mayores realicen actividad física en su tiempo libre a través de talleres deportivos que se realizan tres veces por semana y eventos deportivos masivos.

### Evaluaciones del Programa

Monitoreo Ministerio de Desarrollo Social

**Ejes de Evaluación**

- Objetivo** ★★★★★  
El programa cuenta con un análisis vigente del problema presentado. Su objetivo en concreto y apropiado en relación al problema presentado y a la metodología de intervención propuesta.
- Focalización** ★★★★★  
Si bien el programa incorpora distintas variables para priorizar los lugares donde se va intervenir, las cuales están alineadas con el objetivo, éstas no conforman un sistema que permita dar un orden objetivo de acceso al beneficio. Hay una sobre dimensión de la población objetivo lo que incide en una escasa cobertura del programa, la que alcanza un 0.3%.
- 3a. Medición de Resultados** ★★★★★  
El indicador de resultado permite dar cuenta de manera parcial del logro del objetivo. El indicador de proceso permite hacer un seguimiento parcial de las principales actividades del programa.
- 3b. Logro de Resultados** ★★★★★  
El resultado obtenido el año 2012 sobrepasa la meta establecida, sin embargo, éste resultado no da cuenta del cumplimiento del objetivo

**INFORMACIÓN GENERAL**

**Ley Presupuesto 2013**  
\$592 millones

**Meta Beneficiarios 2013**  
6.480 Personas

➤ **Ámbito de acción**  
Actividad Física y Recreativa

➤ **Tipo de Beneficio**  
Cursos, Charlas y Talleres, Asesorías Técnicas

➤ **Ciclo de vida**  
de 60 a 70 años.

**OTROS AÑOS**  
[2012](#)

Figura 3.7 - “Adulto Mayor en Movimiento”

Se ofrece para cada programa social, su objetivo, focalización, medición de resultados y logros. También muestra de qué servicio depende, si se encuentra vigente, el año de creación, y la url del mismo. Esta información se puede ver en la parte izquierda de la página, por encima de la descripción y objetivo del programa social.

### 3.2.3 Sistema de Argentina

#### Descripción

Argentina ofrece la interacción con mapas para la localización de los recursos, en el cual permite filtrar por diferentes temas. Dentro de lo que define este sitio como temas, se encuentran Mutualistas, Instituciones de Microcrédito, Ferias Permanentes entre otros. La forma de utilización consta del despliegue de todos los recursos en el mapa, y se van eliminando los temas que al usuario deselecciona en los iconos que figuran arriba del mapa interactivo. [36]

La Figura 3.8 muestra la página principal del sitio en la que figura el mapa interactivo.

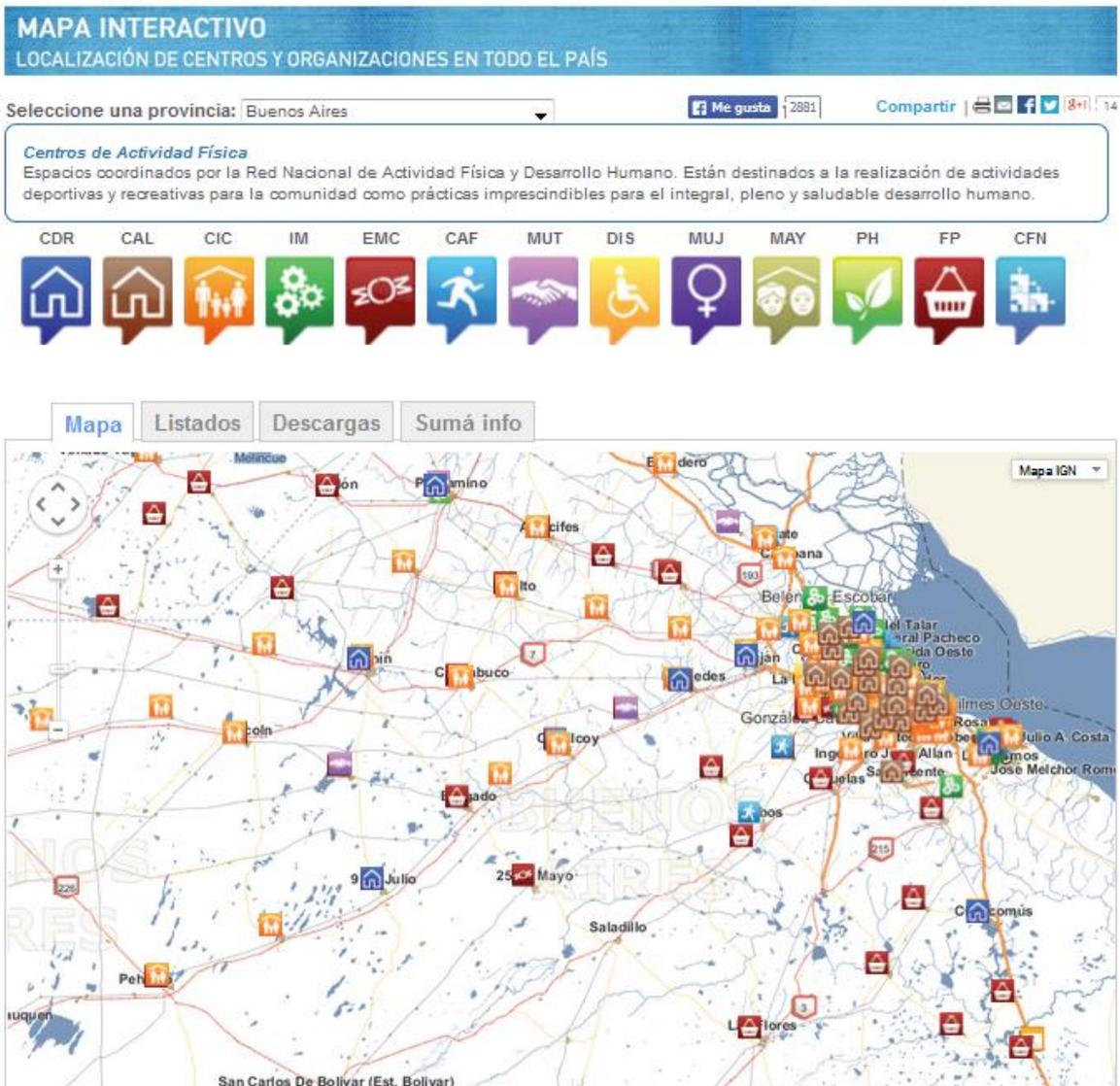


Figura 3.8 - Mapa Interactivo de Sitio Argentino

Luego se puede activar o desactivar las categorías que se despliega en el mapa, a modo de ejemplo se decidió filtrar únicamente por Ferias Permanentes, y el resultado sería el que se despliega en la figura 3.9:

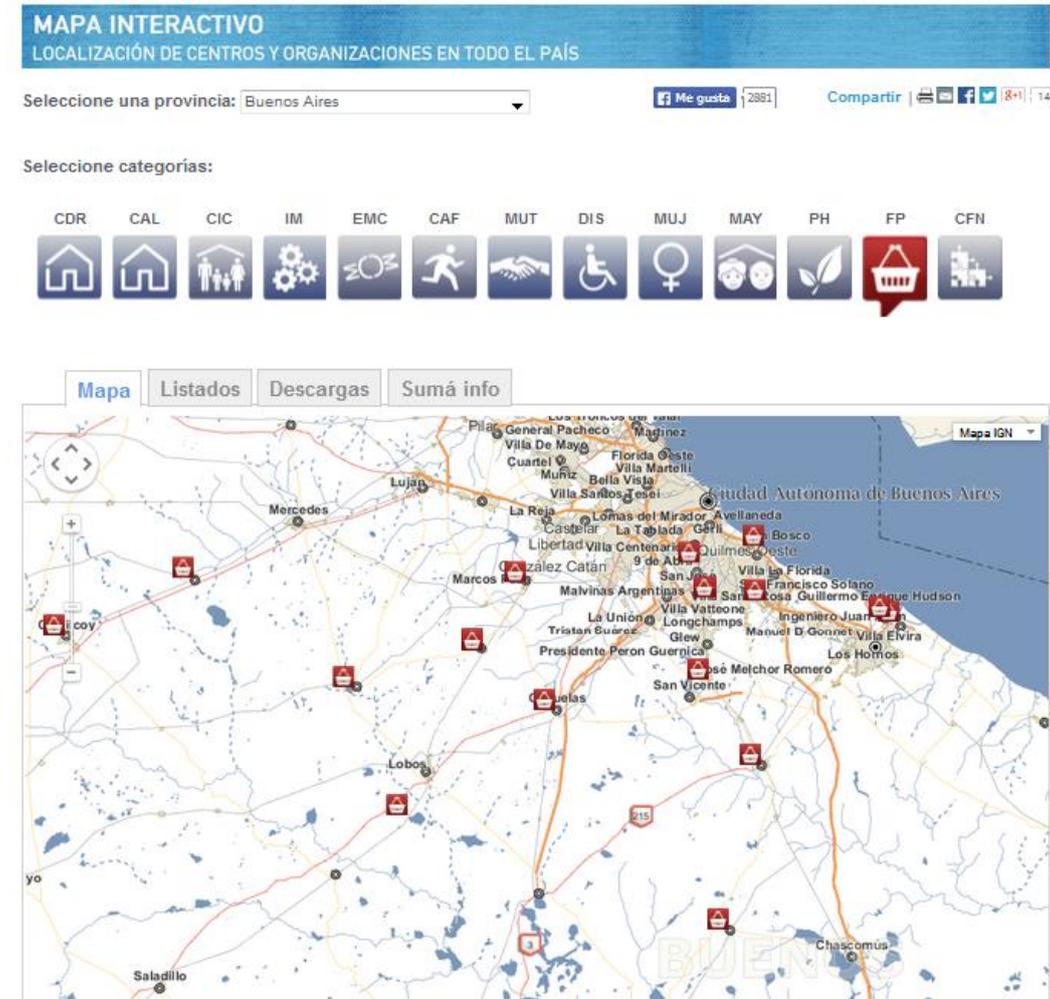


Figura 3.9 – Búsqueda por categoría “Ferias Permanentes”

Si se selecciona la pestaña “Listados” en el menú que se visualiza arriba del mapa, el sistema despliega la pantalla que se muestra en la Figura 3.10:

**MAPA INTERACTIVO**  
LOCALIZACIÓN DE CENTROS Y ORGANIZACIONES EN TODO EL PAÍS

Seleccione una provincia:  Me gusta 2881 Compartir

Seleccione categorías:

CDR CAL CIC IM EMC CAF MUT DIS MUJ MAY PH **FP** CFN

Mapa **Listados** Descargas Sumá info

Filtrá el listado por provincia y categoría  
Utilizó los controles de arriba para filtrar este listado. Versión para imprimir

- Ferías Permanentes (FP)**  
*FERIA PERMANENTE ALMIRANTE BROWN - FRECUENCIA QUINCENAL*  
Predio Municipal (Ruta Nº 4 y Av. Eva Perón) - Almirante Brown (Buenos Aires)  
(011) 4294-6766  
anala\_debernardis@hotmail.com
- Ferías Permanentes (FP)**  
*FERIA PERMANENTE ARRECIFES - FRECUENCIA QUINCENAL*  
Plazoleta de la Estación de la Juventud. (Av. Sarmiento y Av. Dr. Merlazzino) - Arrecifes (Buenos Aires)  
(02478) 451313  
dirprod@arrecifes.mun.gba.gov.ar
- Ferías Permanentes (FP)**  
*FERIA PERMANENTE BALCARCE - FRECUENCIA SEMANAL*  
Plaza Balcarce (Calle 17 y Calle 18) - Balcarce (Buenos Aires)  
(02286) 424007

Figura 3.10 Listado de Ferías Permanentes

Luego si se decide cambiar a la pestaña de descargas encontramos el siguiente listado, lo que permite exportar la información en distintos formatos (Figura 3.11).

**MAPA INTERACTIVO**  
LOCALIZACIÓN DE CENTROS Y ORGANIZACIONES EN TODO EL PAÍS

Seleccione una provincia:  Me gusta 2881 Compartir

Seleccione categorías:

CDR CAL CIC IM EMC CAF MUT DIS MUJ MAY PH **FP** CFN

Mapa Listados **Descargas** Sumá info

Descargá los listados completos

CDR Centros de Referencia	DIS Consejo Federal de Discapacidad
CAL Centros de Atención Local	MUJ Consejo Nacional de las Mujeres
CIC Centros Integradores Comunitarios	MAY Consejo Federal de Mayores
IM Instituciones de Microcrédito	PH Pro Huerta
EMC Emprendimientos Marca Colectiva	FP Ferías Permanentes
CAF Centros de Actividad Física	CFN Consejo Federal de Niñez
MUT Mutuales del Bicentenario	

Toda la información del mapa interactivo está disponible para Google Earth  
 Para visualizarla.

Figura 3.11 – Descarga de listados

La funcionalidad de descarga ofrece la información de cada centro que se busca según la categoría seleccionada, por ejemplo como se despliega en la figura 3.12:



**Ferias Permanentes - FP**  
Espacios de encuentro e intercambio para la comercialización de productos de emprendedores de la Economía Social.

INSTITUCION	PROVINCIA	LOCALIDAD	DIRECCION	CP	TELEFONO	CORREO ELECTRONICO
FERIA PERMANENTE ALMIRANTE BROWN - FRECUENCIA QUINCENAL	Buenos Aires	Almirante Brown	Predio Municipal (Ruta N° 4 y Av. Eva Perón)	B1846ZAB	(011) 4294-6766	anala_debernardis@hotmail.com
FERIA PERMANENTE ARRECIFES - FRECUENCIA QUINCENAL	Buenos Aires	Arrecifes	Plazoleta de la Estación de la Juventud (Av. Sarmiento y Av. Dr. Merlassino)	B2740ZAA	(02476) 451313	dirprod@arrecifes.mun.gba.gov.ar
FERIA PERMANENTE BALCARCE - FRECUENCIA SEMANAL	Buenos Aires	Balcarce	Plaza Balcarce (Calle 17 y Calle 18)	B7620ZAA	(02266) 424007	cmunoz@balcarce.mun.gba.gov.ar
FERIA PERMANENTE BARADERO - FRECUENCIA SEMANAL	Buenos Aires	Baradero	Paseo del Puerto (San Martín y Alte. Brown)	B2942ZAA	(03329) 483627	turismobaradero@yahoo.com.ar
FERIA PERMANENTE BENITO JUÁREZ - FRECUENCIA MENSUAL	Buenos Aires	Benito Juárez	Plaza Ramón Santamarina (Barker, Ordaz N° 550)	B7020ZAA	(02292) 451400	marita.teresa.ricci@gmail.com

Figura 3.12 Detalle del listado de “Ferias Permanentes”

Esta es la primera hoja de un listado en formato pdf, el cual contiene todas las Ferias Permanentes de la Provincia Seleccionada, en este caso Buenos Aires.

### 3.2.4 Conclusiones

#### Argentina

El sitio de Argentina posee un mapa interactivo donde muestra la ubicación de los recursos, además de que permite filtrar los recursos por categorías. Esta funcionalidad maneja el concepto de categorías de manera similar a como se maneja en la GDR.

#### Brasil

El sitio de Brasil al igual que la GDR, ofrece un servicio orientado al área social. Pero a diferencia de la GDR, maneja información estadística de estados y municipios y no los lugares que ofrecen servicios a la población.

#### Chile

El sistema de Chile brinda directamente un buscador de programas sociales. De los programas se muestra una descripción y objetivo del programa, y muestra información relevante del programa. En este sentido Chile maneja una idea similar a lo que se quiere lograr con GDR.

#### Conclusión

Sobre la forma de identificación de los recursos o lugares donde se realizan los programas sociales, se notó que la georreferenciación por medio de mapas no es algo muy utilizado. Realizando una comparación de los sistemas antes descritos, se puede ver que el sistema de Chile maneja directamente la información de los recursos que se ofrecen mientras que los otros sistemas redireccionan a los sitios correspondientes cuando se profundiza en la información. Dado este punto, se separa este sistema del

resto, ya que la GDR tiene como objetivo transmitir la posibilidad de buscar y encontrar la información deseada sin la necesidad de acceder al sitio particular de lo que se pretende averiguar, en la gran mayoría de los casos.

A modo de resumen se muestra en la Figura 3.13 a continuación una comparación sobre las prestaciones que nos interesan sobre estos sitios.

	Brasil	Argentina	Chile
Búsqueda en Mapas	X	X	
Buscador de recursos por texto		x	X
Información sobre recursos		x	X
Información sobre planes sociales		x	X
Información Estadística del Estado	X		

**Figura 3.13 – Comparativa de los sitios**

### 3.3 Casos de Uso

A partir del relevamiento surgieron diversos casos de uso. Los casos de uso más importantes del sistema se encuentran en el diagrama de la figura 3.14. Aquí se pueden apreciar las principales funcionalidades divididas en casos de uso, donde los diferentes actores son los que realizan la acción.

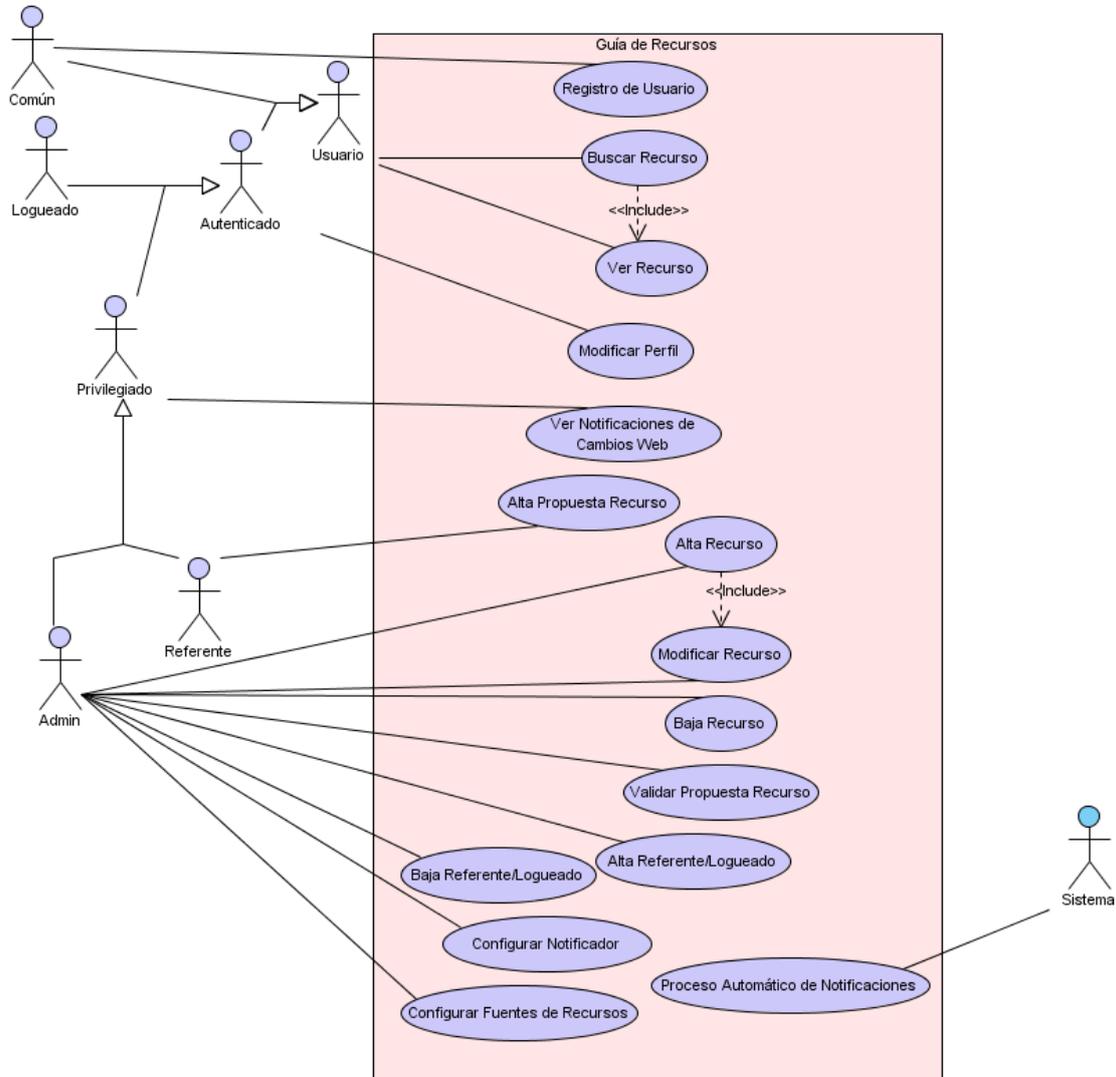


Figura 3.14 – Diagrama de Casos de Uso principales

Cada caso de uso se detalla brevemente a continuación:

#### *Registro de Usuario*

En este caso de uso permite al usuario común registrarse, indicando sus datos personales y su clave de acceso. Luego del registro el usuario pasa a ser un usuario logueado.

#### *Modificar Perfil*

Este caso de uso permite la edición de los datos personales del usuario, su clave de acceso y los recursos y categorías de interés.

### *Buscar Recurso*

Este caso de uso es con el cual se logran las búsquedas de recurso, ya sean por el ingreso de un patrón textual o la elección de las distintas categorías.

### *Ver Recurso*

Este caso de uso es el encargado de mostrar la información completa de un recurso dado. El recurso podrá ser elegido en el caso de uso “Buscar Recurso”.

### *Ver Notificaciones de Cambios Web*

Por medio de esta funcionalidad es que los usuarios administradores y referentes son alertados de los cambios en fuentes web.

### *Alta Propuesta Recurso*

Este caso de uso le permite al usuario referente realizar una propuesta de cambio sobre un recurso. Esta propuesta queda pendiente de validación por parte de un administrador para poder transformarse en un cambio real.

### *Alta Recurso*

Este caso de uso le permite al administrador crear un recurso nuevo.

### *Modificar Recurso*

Este caso de uso le permite al administrador modificar un recurso existente.

### *Baja Recurso*

Por medio de esta funcionalidad es que el administrador puede eliminar un recurso existente.

### *Validar Propuesta Recurso*

En este caso de uso es donde el administrador tomará la decisión de validar, rechazar o validar parcialmente una propuesta realizada por un referente. El caso de la validación parcial implica validar o rechazar partes específicas de la propuesta, pudiendo notar y tomar decisiones sobre cada cambio de forma separada.

### *Alta Referente / Logueado*

Se le ofrece a un administrador la posibilidad de creación de un usuario referente o logueado.

### *Baja Referente / Logueado*

Se le ofrece a un administrador la posibilidad de eliminación de un usuario referente o logueado.

### *Proceso Automático de Notificaciones*

Este caso de uso es el proceso automático que ejecuta el sistema con una determinada frecuencia. Consiste en detectar cambios en fuentes web previamente configuradas.

### *Configurar Notificador*

Este caso de uso proporciona a un administrador la posibilidad de gestionar el proceso automático de detección de notificaciones. Esto consiste en: empezar, parar y cambiar la frecuencia de ejecución, así como también cualquier otro detalle respecto a ese proceso.

### *Configurar Fuentes Web*

Este caso de uso proporciona a un administrador la posibilidad de gestionar las fuentes web del proceso automático de detección de notificaciones. Esto consiste en: crear, modificar y eliminar las fuentes web a detectar, dentro de la modificación es posible asociarlas a los recursos que aportan información.



## 4 SOLUCIÓN PLANTEADA

En este capítulo se describe la construcción de la nueva versión de la GDR, la arquitectura, las tecnologías utilizadas, decisiones tomadas y problemas encontrados.

### 4.1 Arquitectura

La arquitectura del sistema consiste en tres capas: capa de presentación, capa lógica y capa de acceso a datos. De este modo se logra un buen nivel de abstracción para el desarrollo del mismo.

Como se muestra en la Figura 4.1 existe una base de datos la cual es accedida por la capa de acceso a datos. La capa de acceso a datos le brinda los datos a la capa lógica para que trabaje con ellos. La capa lógica provee de lógica de negocio a la capa de presentación para que esta pueda hacer efectivas las acciones del usuario. Finalmente la capa de presentación es quien se encarga de interactuar con el usuario, mostrándole la información y guardando datos temporales que el usuario ingrese o seleccione.

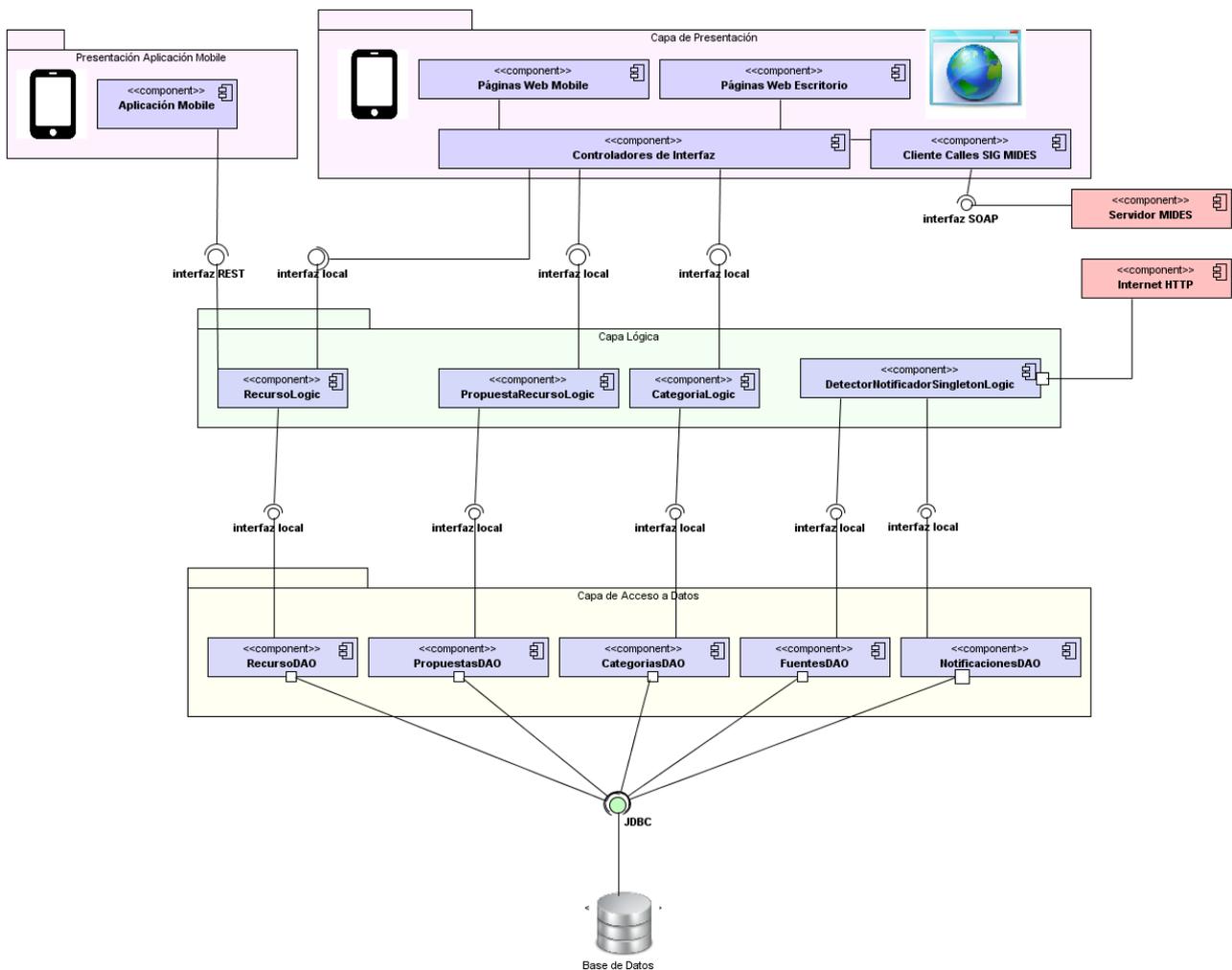


Figura 4.1 – Diagrama de Componentes

### *Base de datos*

Este componente es el encargado de mantener los datos guardados de forma no volátil. Los componentes superiores pueden consultar, ingresar y modificar los datos. La base de datos escogida es PostgreSQL.

### *Capa de acceso a datos*

Dentro de esta capa están los componentes “DAO” que tienen el código de las consultas, inserciones y modificaciones hacia la Base de Datos. Cada uno de ellos está directamente relacionado con una Entidad clave del modelo, de este modo el código de la aplicación queda modularizado.

A nivel tecnológico son Stateless Enterprise Java Beans (EJB) de sesión, los cuales ya tienen incorporado propiedades transaccionales para las modificaciones de la Base de Datos. El servidor de aplicaciones crea instancias de dichos elementos para atender las necesidades de los usuarios, según la cantidad de usuarios conectados.

### *Capa de negocio*

En esta capa se encuentran los componentes “Logic” que tienen el código de la lógica de negocio. Se conectan con los DAO para trabajar con los datos. Cada uno de ellos está directamente relacionado con una Entidad clave del modelo, de este modo el código de la aplicación queda modularizado.

A nivel tecnológico son Stateless EJBs de sesión, los cuales ya tienen incorporado propiedades transaccionales para las modificaciones de la Base de Datos. El servidor de aplicaciones crea instancias de dichos elementos para atender las necesidades de los usuarios, según la cantidad de usuarios conectados.

Los componentes “Logic” inician las transacciones y los “DAO” se unen a estas, si hay un error en cualquiera de los dos componentes involucrados todo volverá al estado inicial. El nivel de aislamiento es configurable.

### *Capa Lógica - DetectorNotificadorSingletonLogic*

Este componente es el encargado de detectar cambios de las fuentes web en internet. Luego dados los cambios encontrados se crean las notificaciones hacia los usuarios correspondientes. Por ese motivo es que se comunica con FuentesDAO y NotificacionesDAO.

El detectorNotificadorSingletonLogic es un EJB que implementa patrón de diseño Singleton e inicia su ejecución de forma automática.

### *Capa de Presentación - Controladores de interfaz*

Estos componentes son los encargados de nutrir a la páginas web de información así como también reaccionar a eventos del usuario sobre las páginas: clicks, selecciones, etc.

A nivel tecnológico están implementados con los frameworks de Context Dependency Injection (CDI) y JSF Managed Beans. JSF y CDI <sup>2</sup> ofrecen Managed Beans con diferentes características, entre ellas está lo que se llama “scope”.

---

<sup>2</sup> “Context and Dependency Injection (CDI)” se incorpora a la especificación Java EE 6. Permite que cualquier JavaBean sea utilizado como un JSF Managed Bean, incluyendo beans de sesión como Stateless, Statefull y Singleton. [59]

El “scope” es el tiempo de vida del Bean, Este tiene varias opciones como por ejemplo, la duración de la sesión del usuario o la duración total de la aplicación entre otros.

Los Controladores de Interfaz se comunican con los componentes “Logic” con el fin de recuperar información y también para el procesamiento de la misma.

#### *Páginas Web Escritorio*

Estas son las páginas web para ser accedidas desde PC's de escritorio. Están construidas con el framework JSF, por lo tanto el servidor las transforma a HTML y JavaScript en cada petición de un usuario determinado. Adicionalmente a JSF están basadas en el framework PrimeFaces que trabaja sobre JSF. Primefaces sirve de ayuda para el desarrollo y mantenimiento de las páginas.

#### *Páginas Web Mobile*

Estas son las páginas web para dispositivos móviles como por ejemplo celulares. Están construidas con el framework JSF, al igual que las de escritorio el servidor las transforma a HTML y JavaScript. Para este caso se utiliza el framework PrimeFaces Mobile.

#### *Aplicación Mobile*

Este componente es la aplicación que se ejecuta dentro de un dispositivo móvil que se comunica con el servidor. A nivel tecnológico la comunicación es lograda mediante WebServices livianos REST.

#### *Cliente calles SIG MIDES*

Este componente es el encargado de comunicarse con el servicio provisto por el MIDES. Este servicio consiste en un listado de calles y elección de las mismas, donde se devolverá una geo-localización de la combinación de calles (esquina) o calle y número de puerta ingresado. Tecnológicamente, esto implica consumir un Webservice SOAP provisto por el equipo de informática del MIDES.

En la Figura 4.2 se muestra cómo será la distribución de los distintos artefactos de software en la implantación del sistema. Un nodo tendría instalada la Base de Datos y otro nodo el servidor GDR JBoss. Los demás nodos son externos y se comunican por medio de WebServices.

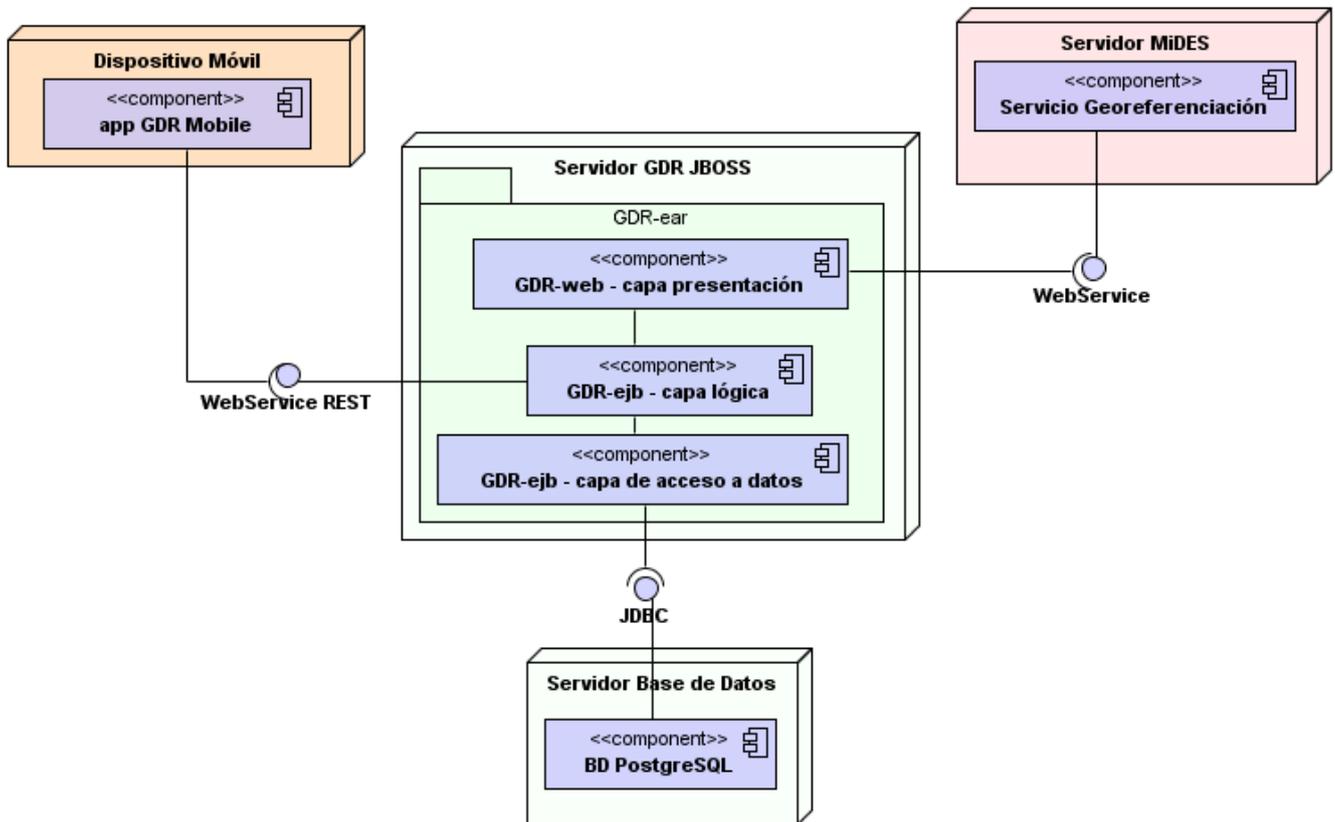


Figura 4.2 – Diagrama de Despliegue

## 4.2 Principales Funcionalidades

Se presentan las principales funcionalidades que conforman la solución planteada.

### 4.2.1 Detección automática de cambios en la web

Para ayudar a la actualización por parte de los administradores, se les brinda una ayuda para que tengan certeza de cuáles son los recursos que cambian su información a medida que pasa el tiempo. Esta funcionalidad consiste en un envío periódico de notificaciones sobre la detección de cambios de las páginas web relacionadas a información de los recursos.

Esta tarea será exitosa si es que se pueden solucionar los obstáculos que la web conlleva, ya que también es posible confundir un pequeño cambio con un cambio real y enviar muchas falsas alarmas. Por ejemplo podría suceder con una página web que muestra la fecha (Cambia diariamente). También existen muchos sitios que generan contenido dinámico que varía según el momento que se consulte, existen además imágenes, archivos flash, pdf's. Para evitar caer en esos obstáculos se investigaron diversas formas de detectar los cambios.

Esta sección del documento se dividió en dos partes. La primera parte es la implementación del detector el cual está hecho con un “parser” HTML. La segunda parte trata sobre la configuración del proceso automático que está hecho con un EJB Singleton.

#### *Detector de cambios*

La solución planteada fue por intermedio de una API para la plataforma Java SE llamada Jsoup [37]. Jsoup brinda posibilidades de trabajar con una página transformándola en un objeto con muchas posibilidades de manipularlo.

Jsoup provee una opción para filtrar por selectores de CSS [38], por lo que el usuario puede configurar los selectores y de esta manera obtener una selección de la página. De esta manera, el sistema concatena toda la información para luego calcular el MD5 asociado a esa página o fragmento de la misma.

El sistema guarda este hash y luego en el proceso automático se chequea, calculando nuevamente su MD5 para saber si la página cambió o no. Si se considera que la página cambió entonces se le notifica a todo administrador y referente en cuyo perfil esté configurado como monitoreador del recurso o alguna categoría asociada al recurso.

La forma que el sistema notifica a los usuarios y las acciones por parte del administrador o referente, son como se muestra en el diagrama de la Figura 4.3.

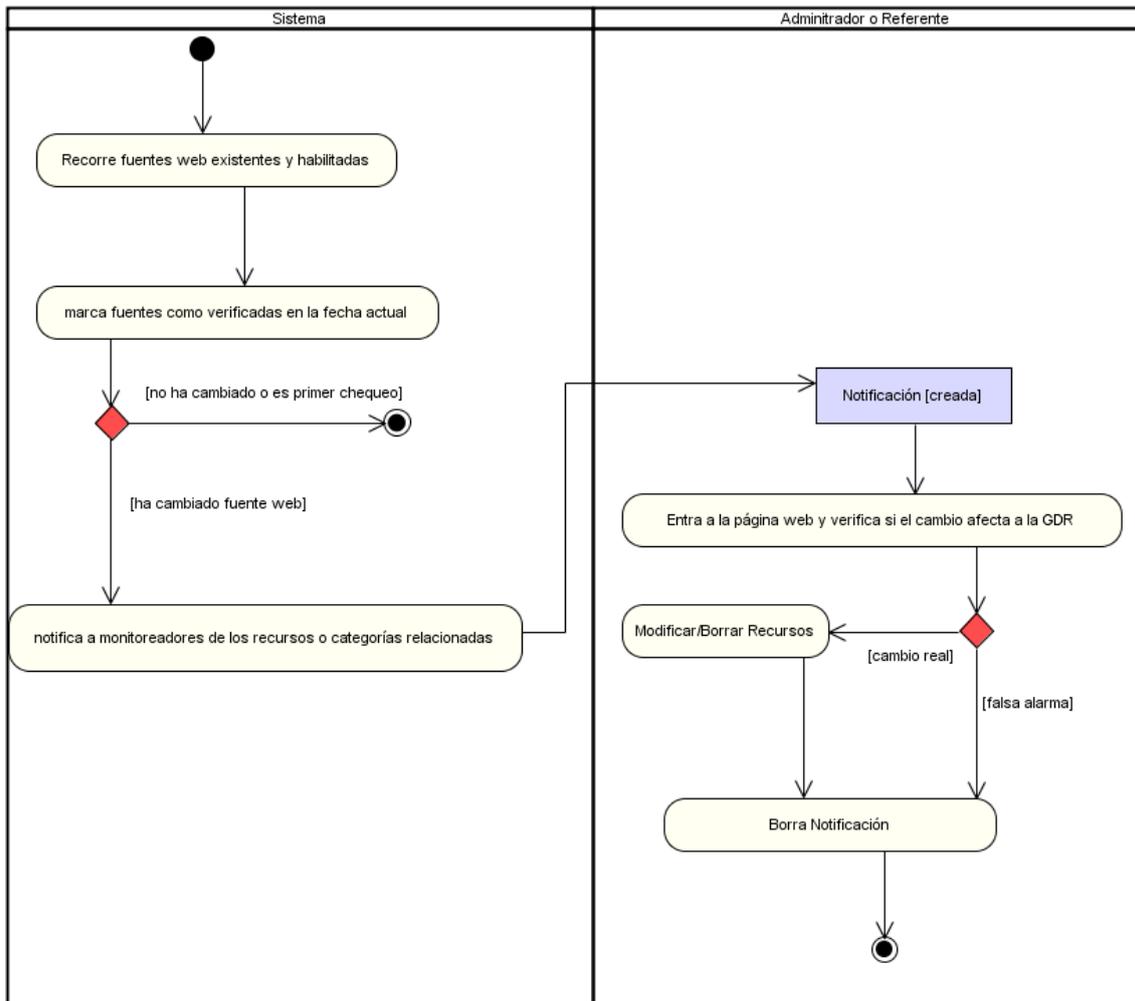


Figura 4.3 Diagrama de interacción

### Configuración del Proceso Automático

Las notificaciones están incluidas en un módulo realiza un proceso automático. Este proceso chequea en forma periódica la existencia de cambios en la web.

La configuración de este proceso se realiza con un EJB Singleton que implementa TimerService. Esta funcionalidad es ofrecida en Java EE 6 para los EJB [39]. La misma permite la planificación de ejecuciones automáticas por tiempo configurable.

Los administradores tienen una interfaz web donde pueden configurar lo siguiente:

- Momento de la próxima ejecución
- Frecuencia con la cual se ejecutará el proceso
- Detener el proceso de control de cambios de las páginas
- Comenzar nuevamente el proceso de control de cambios de las páginas

#### 4.2.2 Mecanismo de actualización de recursos

En este punto se describe el mecanismo propuesto para la actualización de los recursos. Este mecanismo involucra a los roles de usuario administrador y referente. El administrador es el único tipo de usuario que está autorizado a modificar los recursos, ya sea de forma directa o indirecta. El tipo de usuario referente tiene la posibilidad de aportar modificaciones (Propuestas de Recursos) que no serán oficiales hasta que un administrador las verifique y modifique.

Al administrador se le brinda una interfaz de modificación directa. Esta interfaz posibilita la modificación de datos básicos, oficinas y categorías del recurso seleccionado. El Referente tiene permiso de realizar propuestas de recursos en una interfaz similar a la del administrador.

En el diagrama de la Figura 4.4 se muestra cómo es el funcionamiento de una propuesta de cambio.

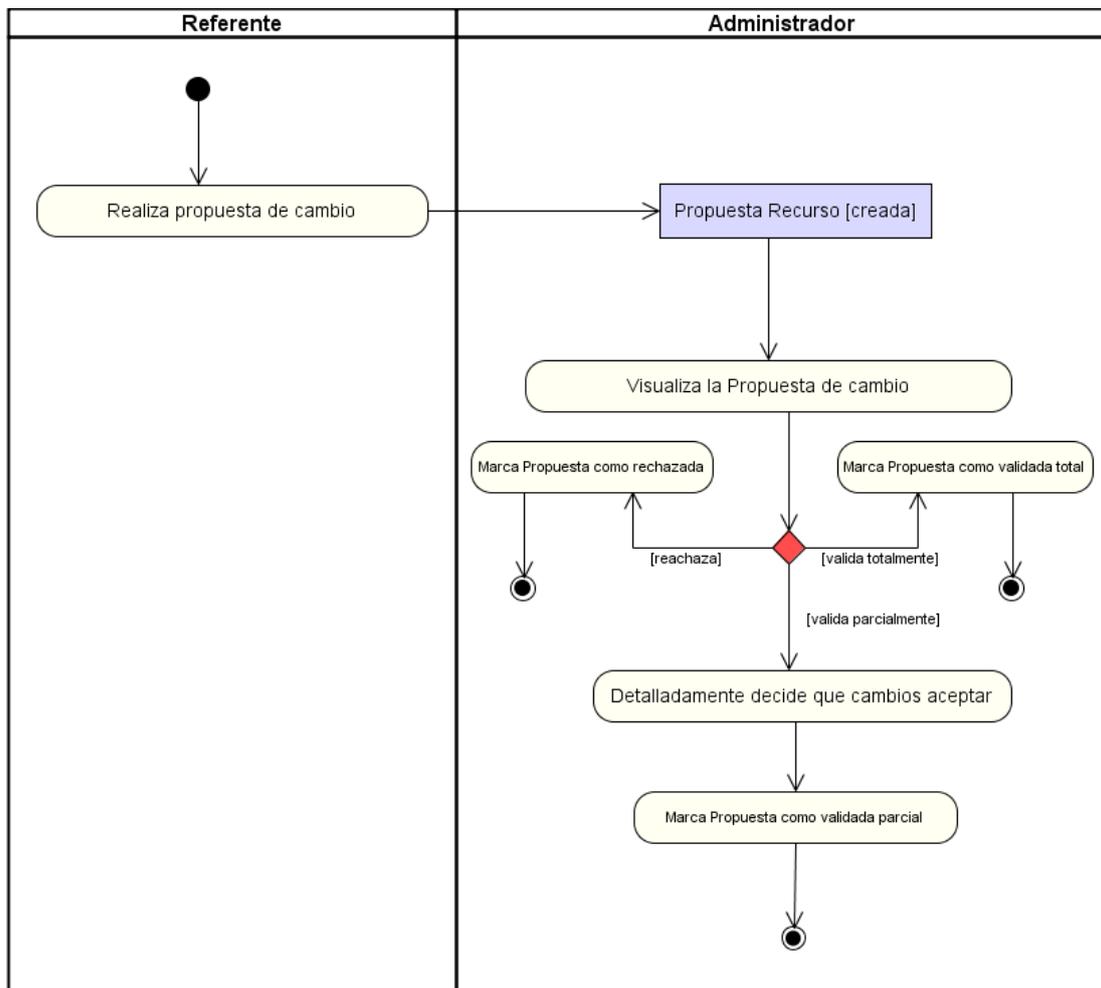


Figura 4.4 – Diagrama de propuesta

Luego de que el usuario referente realiza su propuesta, los administradores son notificados de la propuesta existente. Los administradores pueden visualizar la propuesta de recurso así como el referente que la propuso. Los administradores tienen 3 posibilidades de acción sobre la propuesta: una **Validación Total**, un **Rechazo** y una **Validación Parcial**.

Las posibilidades de validar totalmente y rechazar son opciones de acción rápida. Por otro lado, la decisión de validar de forma parcial la propuesta implica entrar a nuevas pantallas que están en forma de “Wizard”. En ellas se pueden apreciar de forma detallada cada cambio. Por ejemplo en la Figura 4.5, si en la propuesta se modificó el horario de atención, se muestra el horario de atención actual (izquierda) junto al horario de atención propuesto (derecha).

Figura 4.5 – Merging de propuesta y recurso

En resumen, este mecanismo de actualización involucra a todos los usuarios con potestad para realizar cambios en la información de la GDR. A su vez, la autorización para oficializar los cambios es solamente para los administradores. Los permisos de los diferentes usuarios fue la petición exacta de los clientes, relevada en la etapa de requerimientos.

#### 4.2.3 Visualización de la información geográfica

Una de las necesidades identificadas es la de ubicar a los recursos y sus oficinas en el mapa. Para abordar esta necesidad se decidió consumir el servicio que ofrece Google Maps agregando las ubicaciones que se tienen guardadas en la base de datos.

Para facilitar la implementación se utilizó el framework Primefaces (PF). PF trabaja con Google Maps, pero no ofrece la totalidad de las funcionalidades de Google Maps. Igualmente se puede utilizar la API de Google Maps, para extenderse más allá de lo que ofrece PF.

En la Figura 4.6 se aprecia la pantalla inicial de la GDR donde se muestran todos los recursos ubicados en el territorio uruguayo junto con las oficinas. Si dentro del mapa se selecciona uno de los recursos u oficinas se navega hacia el recurso.



Figura 4.6 - Mapa de Pantalla inicial

En el caso de seleccionar un recurso se despliega una vista como en la Figura 4.6, donde se muestra su ubicación junto a la información del recurso.



Figura 4.6 – Ubicación de recurso

En la tabla de oficinas de la pantalla del recurso, si se selecciona una oficina se muestra su localización (Figura 4.7).

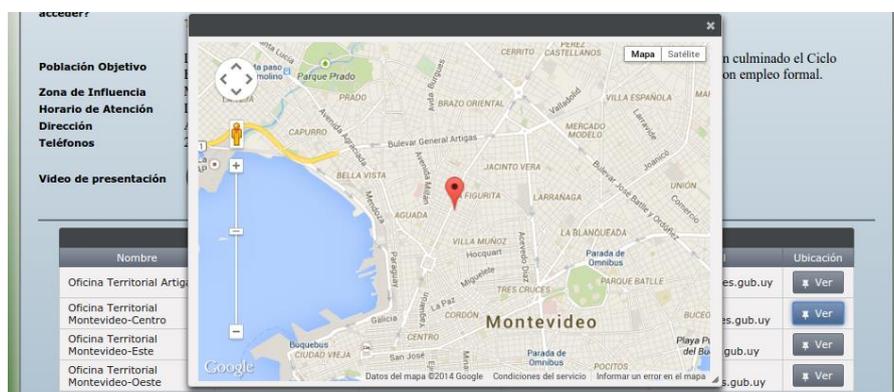


Figura 4.7 – Ubicación de oficina

En la edición de recursos y oficinas, se brinda la opción de indicar su ubicación, además se ofrece la opción de quitar la ubicación.

#### 4.2.4 Búsqueda de Recursos

Se muestra una página inicial con los recursos que ofrece un buscador. Las búsquedas de texto para la GDR deben ser tales que un usuario al recordar el nombre de un recurso, pueda encontrarlo. De esta forma cuando el usuario ingresa un texto similar, encuentra el resultado esperado.

Se mejoró la búsqueda de los recursos utilizando la distancia de Levenshtein, la cual ya viene incluida nativamente en la Base de Datos utilizada. Usando una distancia para comparar textos nos ayuda a no caer en el error de escritura, un tilde o a alguna letra, ofreciendo un servicio de búsqueda más fácil de usar y certero.

Dado que la cantidad de recursos es relativamente baja (cerca de 600 recursos) no se ve un beneficio al utilizar un algoritmo más complejo que la distancia de Levenshtein.

En la Figura 4.8 se muestra la consulta realizada con sus parámetros de configuración. Estos parámetros se escogieron a base de pruebas empíricas.

```
SELECT r FROM Recurso r
ORDER BY levenshtein(LOWER(r.nombre), LOWER(:patronBusqueda),30,1,15),
r.cantVisitas DESC
```

Figura 4.8 – Búsqueda de recursos

En la Figura 4.9 se muestra cómo presenta el sitio la funcionalidad de búsqueda

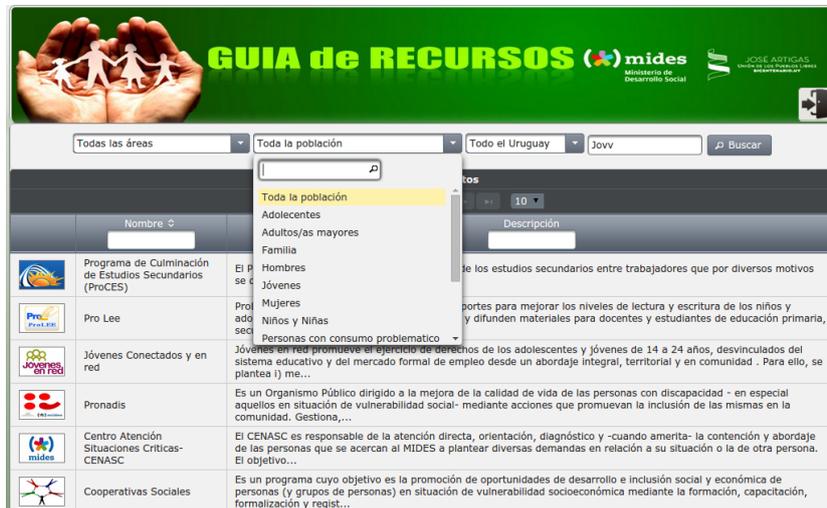


Figura 4.9 – Búsqueda de recursos

#### 4.2.5 Gestión de Categorías

En la GDR un recurso puede pertenecer a una o más categorías que conforman una clasificación taxonómica, la cual permite agrupar a los recursos. Al contar con una categorización de recursos se cuenta con una nueva forma de búsqueda de los mismos. Esto es útil para hacer una rápida referencia a un conjunto de recursos y con ello buscar un recurso específico, así como también describir a determinado recurso.

La resolución jerárquica se modeló como una estructura arborescente, donde cada recurso está asociado a cualquier cantidad de categorías sin restricciones. De este modo, al momento de buscar por una categoría específica, un recurso es encontrado si éste está asociado a la categoría especificada o a una subcategoría de ésta.

También es posible que el usuario especifique más de una categoría al realizar una búsqueda. Esto hace que los recursos encontrados sean los que tienen asociadas todas las categorías especificadas directamente o una subcategoría de ellas. Sobre este tema, para hacer más simple y amigable la interfaz gráfica se decidió permitir especificar solamente una categoría de búsqueda por cada tipo de categoría, o sea que como máximo podrá especificar 3 categorías:

- Área
- Población Objetivo
- Territorio

En la figura 4.10 se encuentran las categorías que se manejan en el prototipo de la nueva versión de la GDR.



Figura4-10 Categorías de la GDR

#### 4.2.6 Dispositivos Móviles

Hablar de diseño de interfaces para aplicaciones móviles, no significa hablar de diseño totalmente enfocado para celulares. No hay que olvidar otros dispositivos portables como son las tablets, netbooks. Todos estos dispositivos poseen diferentes tamaños de pantalla y resoluciones, pero tienen algo en común, poseen una pantalla táctil y no necesariamente se utiliza un mouse para interactuar.

La interacción se hace seleccionando directamente en la pantalla con otro elemento como por ejemplo: dedos y lápices especiales. Esto hace diferente la forma de interactuar, no solo porque en general las pantallas son más chicas, sino que tampoco se dispone del clic-derecho, ni de un puntero que tiene la posibilidad de señalar con mucha precisión un lugar de la pantalla.

En general, un dispositivo móvil cuenta con una velocidad de acceso a internet menor a una PC. Esto es porque normalmente el acceso es por red inalámbrica, aunque en la actualidad están convirtiéndose en redes con más velocidad de transferencia de datos.

En conclusión, para ofrecer una funcionalidad optimizada en dispositivos móviles es mejor realizar un diseño minimalista con textos cortos, botones grandes y tamaño de letras grandes. No debe tener scrolling lateral, debe tener alto contraste de colores y en lo posible minimizar la carga de datos que se le solicitan al servidor.

Si bien nuestra arquitectura posee un controlador que se encarga de las funcionalidades que tendrán los dispositivos móviles, se decidió que la implementación de una aplicación nativa para dispositivos móviles queda por fuera del alcance. [40]

Se decidió implementar una interfaz web mobile, la cual fue implementada totalmente con PrimeFaces Mobile (PFM). El sitio mobile es accedido mediante una url distinta a la url principal de la GDR. El sistema detecta automáticamente si el acceso al sitio es mediante un Celular o un Tablet y redirige hacia la url correspondiente a la versión mobile. Además, aunque se esté en un Celular o Tablet siempre se permite ingresar la clásica interfaz de escritorio.

En esta versión de la interfaz mobile de la GDR se incluyeron dos funcionalidades principales:

- Búsqueda de Recursos:
  - Simple (solo ingresando texto)
  - Avanzada (se puede ingresar texto y filtrar por categorías)
- Consulta de recursos

La Figura 4.11 muestra cómo se visualiza el sitio de la GDR en un móvil.



Figura 4.11 – Captura de interfaz web Mobile

#### 4.2.7 Autenticación y Autorización

La autenticación y la autorización son conceptos de ingreso al sistema y a sus funcionalidades. Dependiendo si se cuenta con una comprobación de identidad y si las funcionalidades del sistema se encuentran restringidas o no para el determinado tipo de usuario en cuestión.

Para ambas acciones (autenticación y autorización) se decidió utilizar Apache Shiro que es una herramienta libre extensible y simple de usar. Además ofrece el manejo de la sesión web del usuario de forma automática y transparente.

Se implementaron dos módulos un frontend y un backend los cuales brindan funcionalidades según el rol del funcionario.

En el módulo frontend se ofrecen tres tipos de funcionalidades: públicas, para usuarios logueados y para referentes de campo.

- **Funcionalidades públicas (usuario común):** son las funcionalidades de consultas y búsqueda de recursos. Todo usuario con acceso a internet podrá hacer uso de ellas.
- **Funcionalidades usuarios Logueados:** son las funcionalidades de consultas y búsqueda de recursos que incluye además una vista personalizada según su perfil.
- **Funcionalidades referentes de campo:** Además de las funcionalidades de búsqueda y consulta se proveen todas las funcionalidades de propuestas de recursos, monitoreo de recursos e Información Externa.

Como se necesita los dos servicios de seguridad de Apache Shiro (autenticación y autorización) es necesario implementar un Realm que sea subclase del AuthorizingRealm. A este realm le llamamos GDRRealm.

El Realm es la clase que conecta a Apache Shiro con la información de seguridad de la aplicación, donde también se desacopla el origen de la información con la implementación de autenticación y autorización. La información de seguridad ya sean usuarios, roles o permisos pueden provenir desde una Base de Datos, un EJB, un Webservice, entre otros.

El GDRRealm carga un EJB vía un lookup de JNDI y le pide al EJB la información del usuario que proviene de la Base de Datos.

La información del usuario necesaria es:

- Datos del usuario (si existe con el nombre de usuario especificado)
- Clave y salt

Luego Apache Shiro se encarga de verificar si coincide la clave especificada efectuando el algoritmo indicado en la configuración (shiro.ini)

Este Realm define los métodos para las siguientes dos operaciones abstractas provenientes de las clases ancestras [41]:

- **doGetAuthenticationInfo**  
Esta operación debe retornar la información autenticación. En el caso de la GDR el usuario, el hash de la clave en SHA512 y su salt asociado.
- **doGetAuthorizationInfo**  
Esta operación debe retornar la información de autorización. Retorna un objeto que tiene el listado de los roles asociados, en el caso de la GDR para cada usuario solo existe la posibilidad de tener solo un rol asociado: “superadmin”, “administrador”, “referente”, “logueado”.

#### 4.2.8 Hashing de contraseñas

Para introducir seguridad en el manejo de contraseñas es de buena práctica no guardar las contraseñas en texto plano en la base de datos. Esto evita que cualquier usuario que ingrese a la base de datos, pueda leer la contraseña de un usuario. Como se menciona en el capítulo 2, los algoritmos de hashing son la solución y mejor aún si el hash es con salt.

La solución se hizo utilizando las siguientes librerías:

- El módulo de criptografía de Apache Shiro
- Vt-password

El módulo de criptografía de Apache Shiro se utiliza para obtener el hash de las contraseñas y generar su salt. Se utiliza el algoritmo SHA-512 con 300 mil iteraciones, además se utiliza un salt generado con SecureRandom.

Se recurre a Vt-password para generar las contraseñas aleatorias basándose en reglas preestablecidas.

La clase GDRPasswordManager es la encargada de obtener el hash de claves, generar claves aleatorias junto con su hash, y comparar claves en texto limpio con hashes provenientes de la Base de Datos. En ella reside el código que utiliza funcionalidades de Apache Shiro para realizar su cometido [42].

La clase GDRPasswordManager tiene los atributos de clase que se muestran en la Figura 4.12.

```
//número de iteraciones de hasheo  
private static int numeroIteraciones = 300000;  
//tamaño de pass aleatorio  
private static int tamanoPasswordAleatorio = 8;
```

Figura 4.12 Atributos de clase

La clase GDRPasswordManager implementa la siguiente API:

- ***generarClaveHashSalt() : byte[][]***

Esta operación es útil cuando un usuario crea a otro usuario

**Descripción:** Genera una contraseña aleatoria que cumple con las reglas especificadas, con una sal aleatoria y con un número de iteraciones *numeroIteraciones*

**Retorno:** Array de tamaño 3

1. en primer lugar la clave aleatoria en texto plano (hacer new String(retorno[0]))
2. en segundo lugar el hash de la clave, en bytes crudos
3. en tercer lugar el salt generada aleatoriamente usada para el hash, en bytes crudos

- ***hashearClave(claveTextoLimpio : String) : byte[][]***

Esta operacion es útil al momento de que el usuario cambia la contraseña, o cuando se registra en el sistema

**Descripción:** Dada una clave en texto limpio, obtiene su hash con un salt aleatorio y con un número de iteraciones *numeroIteraciones*

**Parámetros:**

- claveTextoLimpio: clave proveniente del usuario en texto limpio como por ej. un intento cambio de contraseña

**Retorno:** Array de tamaño 2

1. en primer lugar la clave Hasheada y Salada, en bytes crudos
2. en segundo lugar la sal generada aleatoriamente usada para el hash, en bytes crudos

- ***esCorrectaClave(claveComparacionTextoLimpio : String, byte[], claveHasheadaBaseDatosByte byte[], salBaseDatosBytes : byte[]) : boolean***

Esta operacion es útil al momento de que el usuario se loguea, y también cuando cambie la contraseña para comparar la contraseña anterior

**Descripción:** Compara una clave en texto limpio con una clave hasheada, a la clave hasheada se le debe indicar la sal con que se hasheó

**Parámetros:**

- claveComparacionTextoLimpio: clave proveniente del usuario en texto limpio como por ej un intento de login
- claveHasheadaBaseDatosBytes: clave hasheada, generalmente proveniente de una BD
- salBaseDatosBytes: sal con que se hasheo claveHasheadaBaseDatosBytes, generalmente proveniente de una BD

**Retorno:** si es correcto el intento de matcheo, false de lo contrario

### 4.3 Decisiones tomadas

Analizando la realidad, tecnologías existentes, usuarios finales del sistema y lo que engloba el marco del proyecto de grado se tomaron una serie de decisiones.

Las decisiones más relevantes fueron las que se detallan a continuación:

#### 4.3.1 Realizar una interfaz web mobile en vez de una aplicación mobile.

Esta decisión tuvo el sustento de que al implementar una interfaz web mobile no solo se cumple con el requerimiento de permitir el acceso de dispositivos móviles. Sino que además cualquier dispositivo con acceso a internet y un navegador puede acceder al sitio. Simplemente con las prestaciones de interacción propias del dispositivo y al tener una interfaz web específica para este tipo de dispositivos ya es suficiente para las aspiraciones del Cliente.

Con una aplicación mobile, para alcanzar la generalidad que se tiene con la interfaz web, se deben implementar soluciones para los distintos sistemas operativos que poseen los dispositivos. Una de las desventajas, es que el acceso al GPS del dispositivo no es tan directo en el caso de la interfaz mobile. Y en el caso de la aplicación mobile al usuario se le piden permisos una sola vez al momento de la instalación de la aplicación. Este recurso nos brinda la posibilidad de realizar búsquedas georreferenciadas al usuario según la ubicación que se encuentre.

#### 4.3.2 La implementación de ubicación geográfica utilizando Google Maps y no integrando con el sistema mapas que maneja el MIDES.

Cuando se realizó la integración de este componente en el sistema, se tuvo problemas para consumir el Web Service que ofrece el MIDES. Por temas de tiempos y al ser un prototipo, se decidió utilizar Google Maps como alternativa rápida.

#### 4.3.3 Usar parser Jsoup para la detección de cambios

Para la implementación de la detección de cambios se investigaron diferentes opciones, las cuales algunas fueron descartadas por ser poco efectivas y otras por no ser gratuitas. A continuación se detallan las opciones que se manejaron:

- *HTTP-Last modified y HTTP-GET Condicional* El protocolo HTTP da la opción de incluir en la respuesta un atributo que indica la fecha y hora de la última vez que fue modificado un sitio web que se quiere acceder. [43]. Esto es lo más adecuado si la internet fuera perfectamente estándar, pero existen muchos servidores web que no ofrecen estas facilidades o lo tienen deshabilitado. Como por ejemplo la página del BPS y el MIDES no ofrecen ni el Get-condicional y el last-modified.
- *Servicios con funcionalidades de notificación a cambios de sitios web.* Al buscar la oferta de servicios que ofrezcan esta funcionalidad, se notó que no eran gratis ni tampoco de bajo costo en su momento. Además de que no todos cuentan con una extensión para incorporar a un sistema, sino que son sistemas independientes que manejan la forma de notificación propia, por lo que no todos ofrecen un servicio para incorporar a un sistema existente. Alguno de los sitios que ofrecen una funcionalidad similar, son los siguientes: Site Up Time [44], DotCom Monitor [45]

- **Biblioteca para Java** Jsoup es una librería de java que sirve para trabajar con páginas HTML. Provee una rica API con la que se puede extraer información usando los conocidos CSS selectors, operaciones de jQuery y de los Documents Object Models (DOMs) de los navegadores. Con esta herramienta se pueden comparar pequeñas partes de las páginas de interés para cambios. Además se podría eliminar todo contenido dinámico. [37] [38]

**Conclusión:** Finalmente se tomó la opción de utilizar la biblioteca para Java (Jsoup) la cual tiene más ventajas. No tiene ningún costo económico y con ella se encontró una solución efectiva y personalizable.

#### 4.3.4 Usar distancia de Levenshtein para búsqueda de texto

Para realizar búsquedas de texto, se investigaron extensiones para la base de datos con el fin de mejorar los resultados de búsqueda. Estas extensiones son algoritmos de distancia, los cuales no se limitan a encontrar un texto contenido en otro.

Primeramente se describe qué soporte tiene la base de datos PostgreSQL. Nativamente dicha base de datos ofrece 4 funciones de similitud y distancia para strings, estas son:

- **Soundex** – es un algoritmo fonético, compara el sonido natural para el idioma inglés de las palabras ingresadas con las palabras a comparar, es muy útil pero no para el español ni tampoco otros idiomas.
- **Metaphone** – es un algoritmo fonético igual que soundex que se inventó posteriormente, la motivación de la creación fue mejorarlo ofreciendo resultados más exactos.
- **Double Metaphone** – al igual que los dos anteriores es un algoritmo fonético, fue realizado por el mismo autor que Metaphone, este es una mejora de Metaphone.
- **Distancia de Levenshtein** (también conocida como Edit Distance) – es una función que dadas dos palabras indica la cantidad mínima de inserciones, eliminaciones y sustituciones de caracteres para llegar a ser iguales. Notar que en este caso no importa el idioma de entrada. [46]

Existen varios algoritmos que pueden ser más o menos adecuados dependiendo el problema a resolver. Existe una distancia la llamada distancia de Jaro-Winkler que brinda mejores resultados que Levenshtein para palabras cortas. Lamentablemente no existe un soporte nativo en PostgreSQL, igualmente existe una extensión de PostgreSQL llamada pg\_similarity que ofrece diversos algoritmos conocidos, incluyendo a Jaro-Winkler [47].

**Conclusión:** Considerando efectividad, soporte y resultados empíricos de búsquedas, se tomó la decisión de que la distancia de Levenshtein es la más adecuada. Soundex, Methaphone y Double Methaphone son algoritmos fonéticos que solo sirven para el idioma inglés. Sobre Jaro-Winkler no se encontraron desventajas claras, pero al no encontrarle desventajas a Levenshtein, se decidió no se utilizarlo. Con Jaro Winkler hay que depender de una extensión externa, además, podría llegar a ser difícil o imposible su instalación en openshift. Por lo tanto, queda abierta la opción de cambiar a Jaro-Winkler luego que el sistema esté migrado y se puedan hacer pruebas más rigurosas donde se noten casos fallidos con Levenshtein.

## 4.4 Problemas encontrados

Al contar con experiencia en el desarrollo con las tecnologías utilizadas no se encontraron grandes problemas. A continuación se enumeran problemas encontrados durante la etapa de implementación del prototipo.

### 4.4.1 Versión de JSF

La versión de la implementación de Mojarra JSF 2.1.7 que se encuentra en el conjunto de módulos que ofrece por defecto Jboss 7.1.1, posee problemas en el manejo de formularios grandes entre otros. Este problema se solucionó actualizando a la última versión de Mojarra JSF 2.1 (JSF 2.1.29). Este problema se tuvo que solucionar en el servidor de OpenShift<sup>3</sup>.

### 4.4.2 Driver JDBC del servidor de OpenShift

La versión que maneja OpenShift para la comunicación con PostgreSQL, no funciona correctamente al obtener un arreglo de bytes que no son caracteres. Una posible solución era cambiar la configuración de la base de datos. Otra solución la cual fue la elegida, fue la de actualizar la versión del driver de JDBC del Jboss de openshift.

### 4.4.3 Framework PrimeFaces para Desktop

Esta herramienta es muy completa, pero en algunos casos, cuando se intenta realizar algo muy personalizado, surgen errores. Esto requirió una investigación particular para cada uno de estos errores llevando más tiempo de lo esperado.

### 4.4.4 Framework PrimeFaces para Mobile

El principal problema que se tuvo con este framework, fue la falta de documentación y el poco avance en el desarrollo de esta tecnología. La personalización de la interfaz utilizando esta herramienta, es costosa más aún que para la versión para Desktop, y en este caso la documentación es muy escasa. Esto provocó que no se pudiera extender mucho más de lo que ofrece este.

### 4.4.5 Framework de seguridad Apache Shiro

Apache Shiro es un framework muy completo y simple de usar, pero en su documentación oficial no está del todo completa. Se debe recurrir a artículos externos, foros, etc.

---

<sup>3</sup> Openshift es una plataforma que brinda servicios de aplicaciones en la nube [60]



## 5 GESTIÓN DE PROYECTO

A continuación se describe la evolución del proyecto y del producto desarrollado así como los desvíos y dificultades encontrados. El objetivo de este capítulo brindar al lector, una visión global del proceso transcurrido.

En lo que la fue la evolución del proyecto se identificaron cinco etapas, cada una de ellas marca un hito en el proceso de construcción. Aquí se encuentran las tareas que conformaron este proyecto, ubicadas dentro de la etapa en la que se llevaron a cabo.

### 5.1 Etapas del Proyecto

El proyecto consistió en 5 etapas que se presentan a continuación.

#### 5.1.1 Etapa 1 – Inicio del Proyecto y Análisis Preliminar

El proyecto Guía de Recursos del MIDES tuvo comienzo en abril del 2013.

En primera instancia se coordinó una reunión entre los encargados del proyecto, las tutoras y los clientes, con el fin de presentar y comentar en que se basó la investigación realizada por el InCo sobre lo que en ese momento era la Guía de Recursos del MIDES.

Se coordinaron reuniones con los clientes para empezar a relevar los requerimientos del sitio. Estas reuniones se basaron en dudas, preguntas, propuestas, tormenta de ideas, a partir de eso se realizaba un acta de reunión. Dicha acta posteriormente se les enviaba a los clientes para su validación, ya que se consideró que se podría no haber entendido la misma realidad que ellos describían. Además sirvió para dejar un registro de las funcionalidades que se iban acordando.

A medida que se entendía el objetivo del cliente, se comenzó a investigar qué tipo de tecnología se iba a necesitar. Uno de los intereses tecnológicos que tenía este proyecto era la utilización de la plataforma JavaEE para desarrollarlo. Tomando esto en cuenta y analizando la infraestructura que el MIDES posee, se comenzó a evaluar las herramientas que se podían y no se podían utilizar. El MIDES en particular trabaja con servidores de aplicaciones que dentro de poco quedarán en desuso (Jboss5). Dentro de las posibilidades de salto tecnológico se decidió utilizar un Jboss7. Se pensó en utilizar un servidor en la nube, pero teniendo en cuenta que al ser una institución pública, sería difícil que a corto plazo se realizara ese cambio en el resto de los sistemas que maneja el MIDES.

Se realizaron una serie de reuniones con los clientes, en las cuales se determinaron las funcionalidades centrales que la nueva Guía de Recursos debía poseer. En paralelo se investigaron sistemas con objetivos similares para poder ofrecer nuevas ideas para la GDR, que luego junto a los clientes se decidía si era conveniente su incorporación.

Se realizaron investigaciones para poder determinar si se podían satisfacer ciertas funcionalidades. Entre ellas se destaca la autonomía del sistema para poder detectar cambios en sitios web.

Una vez que se notó que se contaba con las herramientas para poder satisfacer todos los requerimientos se pasó a la Etapa 2.

### 5.1.2 Etapa 2 – Investigación Tecnológica y Bosquejos de Interfaz

En esta etapa se profundizó en las tecnologías que se venía investigando, pero ahora con el fin de decidir cuál se utilizaría realmente.

Primeramente se decidieron cuáles serían las tecnologías base, basándose en:

- Sugerencias de las tutoras basadas en su experiencia
- Confiabilidad y conocimiento que los integrantes del proyecto tienen sobre algunas tecnologías
- Investigación de otras alternativas tecnológicas.

En segundo lugar, teniendo fijas las tecnologías base, se estudiaron y compararon otras tecnologías para la resolución de problemas más específicos, entre ellos:

- Detección automática de cambios en sitios web
- Búsquedas de texto
- Presentación de información geográfica
- Interfaz para dispositivos móviles
- Hashing de contraseñas
- Autenticación y Autorización

El descubrimiento de la existencia de estas tecnologías fue en muchos casos gracias a búsquedas en internet, consulta con colegas y también en experiencia laboral propia de los integrantes del proyecto. En todos los casos en reuniones con las tutoras se comunicó, se debatió y se obtuvieron sugerencias para refinar las tecnologías a utilizar.

Es importante dejar claro que en todos los casos siempre se optó por tecnologías de uso libre, porque en algunos casos existían soluciones pagas las cuales fueron descartadas por tratarse de una solución académica.

Como la interfaz gráfica es un área clave para el éxito de proyecto, se realizó un diseño tentativo sobre cómo sería la interfaz de las funcionalidades más importantes, con el fin de manejar un estándar de interacción que el cliente considerara más apropiado. Esto sirvió tanto cuando fue mostrado a las tutoras como cuando se presentó al cliente, ya que se pudo mejorar la interfaz para que pudiera ser más intuitiva.

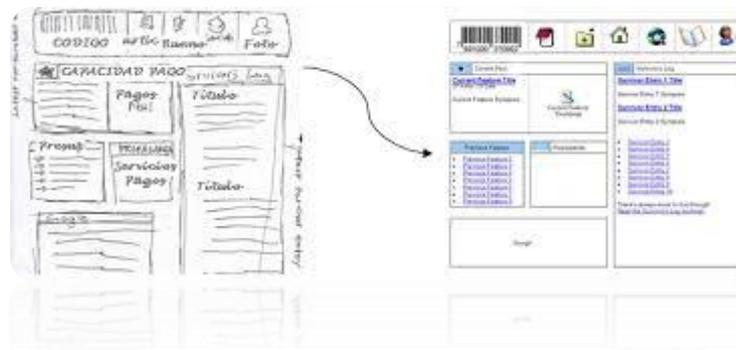


Figura 5.1 -Mapeo de los bosquejos

Otra de las ventajas de haber realizado estos bosquejos, fue que en una etapa temprana se definieron los datos que el usuario debía ingresar y consultar.

Una vez que se definieron los requerimientos y se contaba con un avance en la investigación de tecnologías se pasó a la etapa de elaboración y construcción del producto.

### 5.1.3 Etapa 3 –Elaboración y Construcción del Producto.

En primera instancia se realizó un diseño que permita brindar una solución eficiente. Para esto se realizaron reuniones con las tutoras en donde se debatían las posibilidades que se tenían y cómo se pensaba resolver la problemática.

A raíz de estas reuniones se pudo mejorar la solución, como por ejemplo una de las principales mejoras incorporadas, fue la forma en que se resolvía la jerarquización de las categorías.

En forma paralela se comenzó la escritura del documento de especificación de requerimientos. Se detalló minuciosamente los requerimientos más importantes para el cliente.

Se comenzó la documentación de las tecnologías investigadas, así como las ideas concluidas a partir de los sitios web similares.

Una vez que se contó con un prototipo que tuviera todas las funcionalidades más importantes, se coordinó una presentación de avance con presencia de los clientes. Dando como inicio a la etapa 4.

### 5.1.4 Etapa 4 – Presentación de Avance y Mejora del Producto

En esta etapa se realizó la preparación de la presentación, así como una mejora de interfaz del prototipo. Esta presentación tuvo lugar en el salón de posgrado del InCo (Facultad de Ingeniería), el día 7 de Octubre del 2014, en la cual se contó con una aprobación total de las funcionalidades desarrolladas.

Surgieron sugerencias en la forma de despliegue de información. Se tomó nota de todas las sugerencias de interfaz y de interacción, así como de la incorporación de nuevas funcionalidades. Se acordó realizar las funcionalidades más importantes que fueran rápidamente incorporables.

Esta presentación tuvo el público de las tutoras, investigadores de la Facultad de Información y Comunicación, los clientes y un equipo de informática del MIDES.

Se acordó además brindarle al MIDES una Máquina Virtual para que puedan instalar este prototipo de forma provisoria.

### 5.1.5 Etapa 5 – Cierre del Proyecto, Documentación, y Presentación Final

En esta etapa se cerró el producto, contemplando todas las sugerencias propuestas de la presentación de avance, además de la incorporación de algunas funcionalidades.

Se construyó una máquina virtual la cual contiene una versión de la Guía de Recursos para que pueda ser instalada en cualquier ambiente.

Se desarrollaron las principales secciones de este documento, y se generó toda la información a entregar.

## 5.2 Desvíos y dificultades

A lo largo del proyecto enfrentamos a una serie de imprevistos que provocaron directamente un atraso en la realización del Proyecto. Se determinaron dos grandes áreas en este aspecto, la primera en la cantidad de recursos y la ausencia de alguno de ellos por un lapso de tiempo, y la otra debido a la utilización de la última tecnología para la elaboración y construcción del producto.

En primera instancia, este proyecto tenía como cantidad recomendada de estudiantes tres personas y asumimos este compromiso solo dos. Esto implicó que el camino crítico del proyecto abarcara casi todas las etapas, entonces si uno de los integrantes no puede dedicar tiempo al proyecto, provoca un atraso.

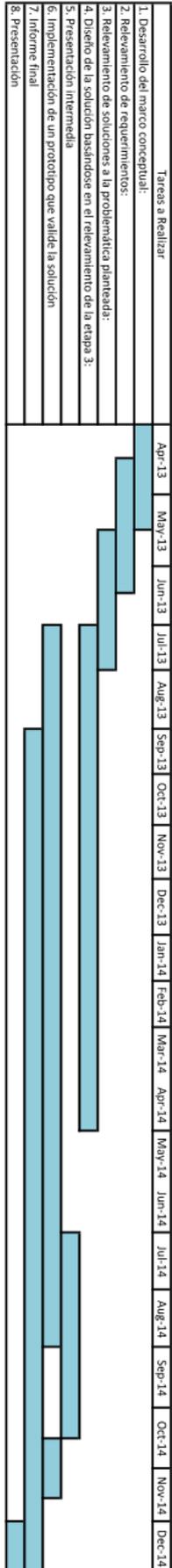
Las principales situaciones que provocaron el atraso del proyecto fueron:

- Julio 2013 - Preparar y rendir Exámenes.
- Diciembre 2013 – Preparar y rendir Exámenes.
- Febrero 2014 – Intervención Quirúrgica de uno de los estudiantes.
- Marzo 2014 – Abril 2014 – Problemas con el equipo informático.
- Junio 2014 - Intervención Quirúrgica de uno de los estudiantes.
- Julio 2014 – Prepara y rendir último Examen
- Octubre 2014 – Problemas Personales de uno de los estudiantes.
- Octubre 2014 – Mudanza de uno de los estudiantes.
- Diciembre 2014 – Mudanza de uno de los estudiantes.

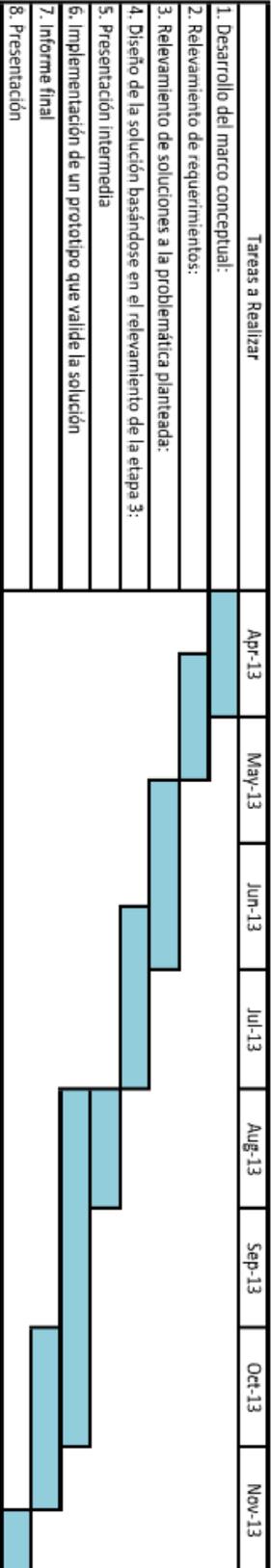
A su vez se determinaron retrasos en la implementación de ciertas tecnologías. A modo de ejemplo las carencias de documentación en los siguientes frameworks: Primefaces mobile, Apache Shiro.

A continuación se muestran los diagramas de Gantt correspondientes a la planificación inicial por parte de la propuesta del proyecto de grado y cómo realmente lo concretamos.

**Forma de realización:**



**Propuesta Inicial:**





## 6 VALIDACIÓN CON USUARIOS

En los clásicos procesos de desarrollo siempre se dejan los detalles de interfaz para el final. Esto hace que se pase directamente a la implementación de la interfaz sin haberle preguntado al usuario cuáles son sus preferencias.

En este capítulo se muestra el proceso de validación llevado a cabo. No es estrictamente el proceso centrado en el usuario, sino que es una versión mucho más resumida pero que hace partícipe a uno de los usuarios finales, en este caso el administrador de la GDR.

Al tener una respuesta del usuario al inicio del proceso, se termina obteniendo una ganancia de tiempo a largo plazo. De ese modo se evita el doble trabajo de implementación por cambios surgidos luego de finalizado el producto.

Considerando lo anterior es que se opta por hacer un diseño de interfaz basado en bosquejos. Para ello se utiliza la aplicación Balsamic Mockups. Siendo un proceso más enfocado en el usuario que un proceso tradicional.

### 6.1 Herramienta utilizada

Balsamiq Mockups es una herramienta que permite realizar Wireframes para webs fácilmente. Un Wireframe (aplicado a la web) es una representación esquemática de la solución que se llevará adelante, sin entrar en etapas posteriores como el diseño gráfico o la programación web. Permite acordar con el cliente aspectos clave de la solución a desarrollar, como la distribución general de los elementos, sus jerarquías y la navegación de los mismos. [48]

Balsamic mockups ofrece una forma fácil y rápida de realizar bosquejos de interfaz con un mediano nivel de detalle. Además permite asignar a un bosquejo una navegación a otro, y luego existe una opción de previsualización en pantalla completa. En esa previsualización se puede ver claramente el bosquejo y navegar a otros bosquejos haciendo la experiencia similar a la futura interfaz, esto sirve para presentarlo y validarlo.

Ésta herramienta no es gratuita pero ofrece una aplicación web “trial”, esta versión fue la que se utilizó.

### 6.2 Algunos bosquejos

Se focaliza principalmente el trabajo en la interfaz web de administración la cual es la más completa, los otros perfiles de usuario tienen una interfaz más acotada.

Se muestran a continuación algunos de los bosquejos más significativos:

#### **Página Inicial**

En la Figura 6.1 se puede apreciar la pantalla disponible para todos los usuarios sin haber ingresado al sistema, aquí se tiene un listado con los recursos donde pueden hacer una búsqueda por categorías, una búsqueda simple o una búsqueda avanzada.

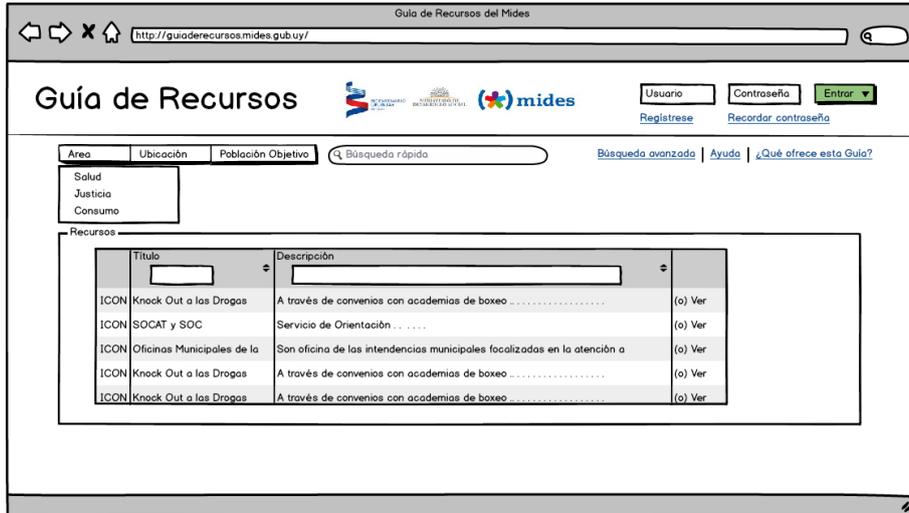


Figura 6.1- Bosquejo de página inicial GDR

### Página de inicio del usuario Administrador

En la Figura 6.2 se le muestran al administrador todos los recursos, los recursos con propuestas de referentes pendientes de validación y los recursos con notificaciones por cambio de página web.

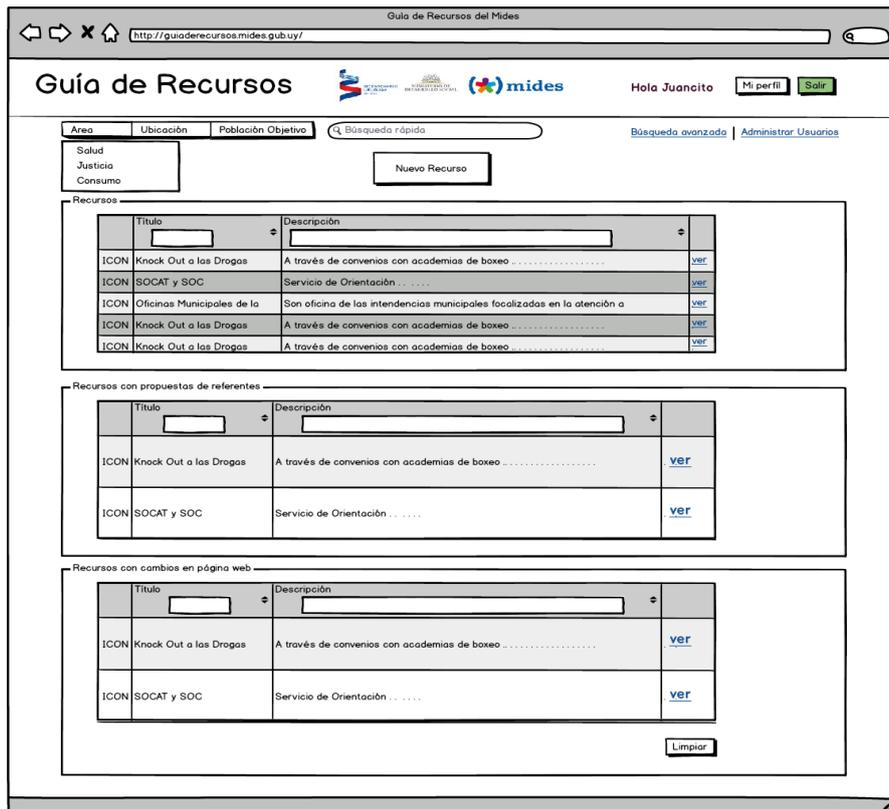


Figura 6.2 – Bosquejo de página inicial de administrador

### Página de validación de propuesta de referente

En la figura 6.3 se muestra a la izquierda el recurso real y a la derecha la propuesta de algún referente de campo. Los administradores quieren tener el control total sobre los cambios de los recursos, pero a su vez quieren tener la información lo más actualizada posible. Es por ello que se opta por esta solución, con lo cual se puede realizar una rápida actualización con poco trabajo por parte de los administradores. Se muestran las diferencias específicas para cada campo donde se puede optar por aceptar algunos cambios y otros no, así como también se pueden editar estos cambios.

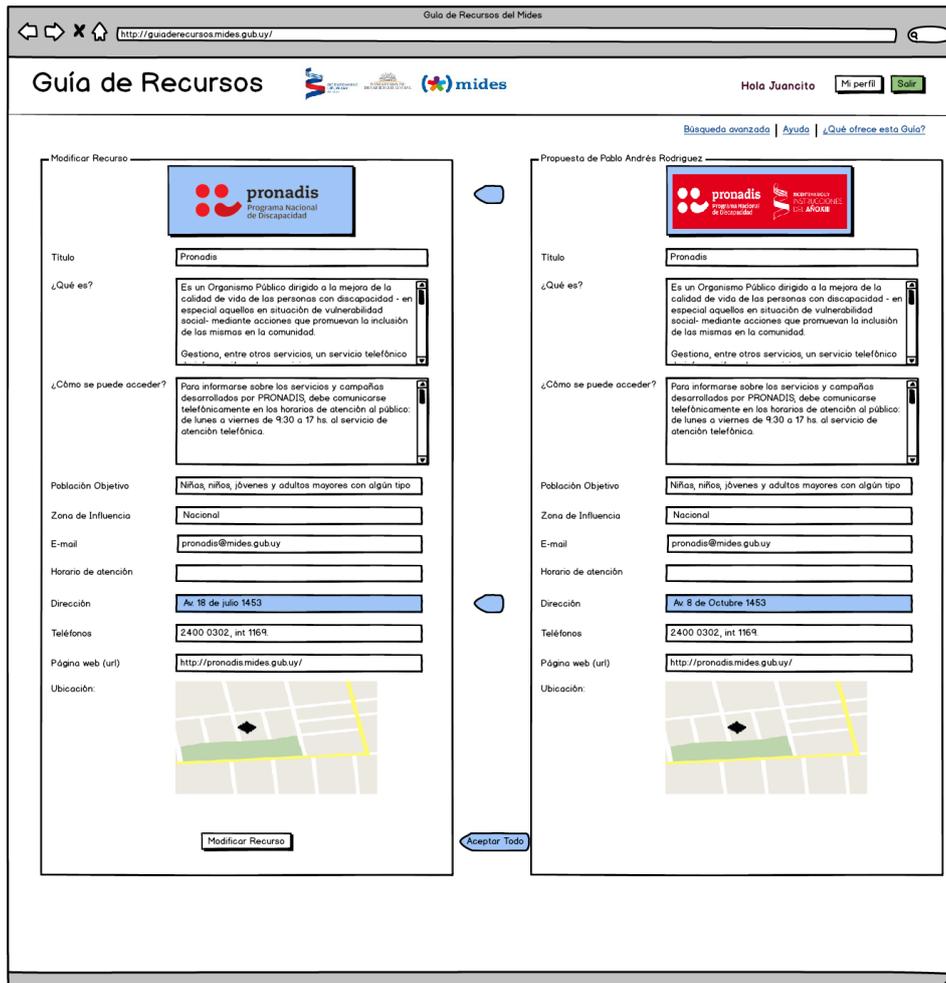


Figura 6.3 – Bosquejo de validación de propuesta de recurso

## Página de modificación de recursos

Esta página está dividida en pestañas:

- En la primer pestaña se aprecia el recurso tal como lo visualiza cualquier usuario
- Modificación de datos básicos
- Agregar, Quitar y Modificar oficina
- Modificar Categorías asociadas

En la Figura 6.4 se muestra cómo se pueden editar los datos básicos, ubicación y logo. En algunos datos básicos se tiene la posibilidad de agregarle formato al texto.

The screenshot shows a web browser window with the URL <http://guiaderecursos.mides.gub.uy/>. The page title is 'Guía de Recursos'. The user is logged in as 'Hola Juancito' with options for 'Mi perfil' and 'Salir'. The navigation menu includes 'Recurso', 'Modificar Datos Básicos', 'Agregar Oficina', 'Modificar Oficinas', 'Quitar Oficina', and 'Modificar Categorías'. The 'Modificar Datos Básicos' tab is active, showing the following form fields:

- Título:** Pronadis
- ¿Qué es?:** Es un Organismo Público dirigido a la mejora de la calidad de vida de las personas con discapacidad - en especial aquellos en situación de vulnerabilidad social- mediante acciones que promuevan la inclusión de los mismos en la comunidad. Gestiona, entre otros servicios, un servicio telefónico de información sobre servicios y programas disponibles para las personas con discapacidad.
- ¿Cómo se puede acceder?:** Para informarse sobre los servicios y campañas desarrollados por PRONADIS, debe comunicarse telefónicamente en los horarios de atención al público: de lunes a viernes de 9:30 a 17 hs. al servicio de atención telefónica.
- Población Objetivo:** Niñas, niños, jóvenes y adultos mayores con algún tipo de discapacidad.
- Zona de Influencia:** Nacional
- E-mail:** pronadis@mides.gub.uy
- Horario de atención:** (Empty field)
- Dirección:** Av. 18 de julio 1453
- Teléfonos:** 2400 0302, int 1169
- Página web (url):** http://pronadis.mides.gub.uy/
- Ubicación:**
  - No especificado
  - Url Infamilia
  - En mapa

A map is displayed below the location options, showing a street grid with a highlighted area in green and yellow. A 'Modificar Recurso' button is located at the bottom of the form.

Figura 6.4 - Modificación de recurso (pestaña modificar datos básicos)

En la Figura 6.5 se presentan los tres tipos de categorías: Áreas, Población Objetivo y territorio. En cada caso se muestran a la izquierda las posibles opciones a elegir y a la derecha se muestra el conjunto de categorías final.

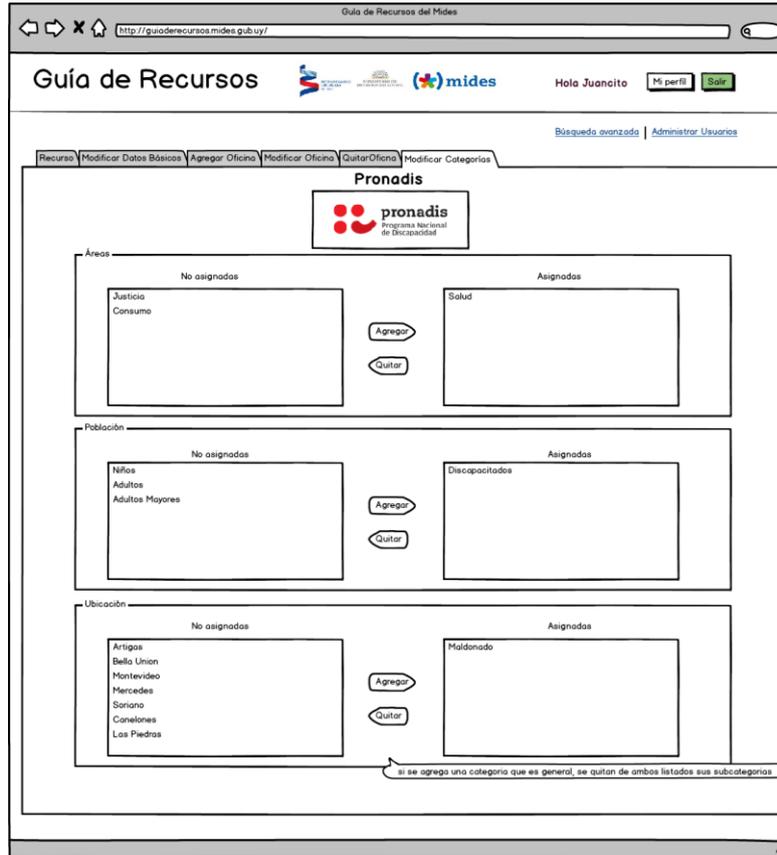


Figura 6.5 - Modificación de recurso (pestaña mostrar categorías)

### 6.3 Resultado de la validación

El resultado fue exitoso ya que esto permitió detectar muchos aciertos en el diseño inicial y a su vez algunas discrepancias. El poder visualizarlo hace que sea más simple la comunicación. Esto se debe a que todos los integrantes de la reunión pueden ver lo mismo al mismo tiempo, y de este modo no se tiene el desfase, desorden y confusión de pensamientos de cada persona.

Una mejor experiencia aún hubiera sido incluir más usuarios y más tipos de usuario, pero ello conllevaba más recursos humanos y una mayor infraestructura, con los cuales en este proyecto no se contaba.



## 7 CONCLUSIONES Y TRABAJOS A FUTURO

### 7.1 Conclusiones

Basándose en los resultados obtenidos, se considera que se alcanzaron los objetivos de este proyecto. Consideramos un balance positivo la realización del mismo, ya que nos vimos enfrentados a un cliente real. La problemática la pudimos analizar y resolver de una manera exitosa.

A lo largo del proyecto se originaron discusiones. Éstas siempre tuvieron un impacto positivo y se llegó a una mejor forma de solución. Por lo tanto, una de las conclusiones a las que se llegó es que la realización de este proyecto permitió un crecimiento profesional por las distintas etapas que se tuvieron que atravesar. Desde un buen relacionamiento personal con el cliente, hasta una discusión específica sobre la utilización de alguna tecnología en particular.

La nueva versión de la GDR ofrece la visualización de los recursos, así como una útil gestión de los mismos por parte de los diferentes usuarios. Se ofrecen formas de búsqueda efectivas y simples, el usuario no debe ser experto para poder usarlas. La información geográfica es mostrada siempre que sea necesaria. Se ofrece una herramienta para la detección de cambios en páginas web externas con el fin de ayudar al mantenimiento de la actualización de los datos de los recursos.

Al no ser un sistema crítico, aunque no se haya pasado por una etapa de verificación rigurosa, la nueva versión de la Guía de Recursos es una herramienta útil para ponerse en producción. Si se decide implantar este sistema, se debe hacer una migración de los datos de la versión vieja.

Sobre las tecnologías utilizadas se concluye que dejaron un vasto aprendizaje a los estudiantes. La experiencia obtenida sirve para hacer un análisis de su utilización en nuevos proyectos. Java EE ofrece una infraestructura muy útil para el desarrollo de aplicaciones empresariales. PostgreSQL es una base de datos muy potente y simple de usar. Apache Shiro es un framework de seguridad que sirve para cualquier tipo de aplicación, ya sea una aplicación web, de escritorio o dispositivos móviles.

Un aspecto importante de este proyecto fue la realización de la validación de interfaz temprana con los clientes, que a su vez son usuarios finales del producto. Se generaron instancias de validación de interacción del sistema, en las cuales utilizando los bosquejos de interfaz se tuvieron resultados destacables. Luego de finalizado el producto se realizó una presentación donde concurrieron los clientes. En dicha presentación se hizo una demostración de la GDR y los clientes mostraron aprobación total.

Dentro de los puntos fuertes de este proyecto se destaca: la detección de cambios web que puede extraerse como solución a otro proyecto y la gestión de cambios de los recursos que fue aprobada por los clientes.

## 7.2 Trabajos a futuro

Dentro del marco del proyecto se resolvieron una serie de requerimientos, lo que hace que hoy la GDR sea un prototipo que consta con las funcionalidades para la resolución a la problemática existente. De igual manera se detectaron una serie de modificaciones necesarias para una eventual mejora del producto.

Estas mejoras son:

- Mejorar la interfaz, en el sentido de que se pueda orientar más al público que en realidad tendrá la mayor utilización de esta herramienta.
- Realizar una integración efectiva con el SIG del MIDES.
- Inicio de sesión con Facebook, Gmail, Twiter, etc.
- Brindar una personalización de la forma en que se despliegan gráficamente los recursos. A modo de ejemplo, ofrecer distintos diseños de páginas de recursos o también dando la posibilidad de realizar la carga de una hoja de estilos CSS para esa visualización del recurso.

Se realizó una presentación intermedia para los clientes, aquí se les mostró la nueva GDR finalizada y se les invitó a aportar nuevas ideas. La presentación fue muy positiva y con gran aceptación por parte del cliente.

De esta presentación surgieron algunas sugerencias, de las cuales algunas quedaron por fuera del alcance:

- Auditado de los cambios realizados por los usuarios
- Exportación de datos del sistema a un formato manejable por plantillas electrónicas por intermedio de la aplicación y no por la base de datos.

## 8 REFERENCIAS

- [1] «Guía de Recursos del MIDES,» [En línea]. Disponible: <http://guiaderecursos.mides.gub.uy/mides/index.jsp>. [Último acceso: 1 junio 2013].
- [2] «Innova Portal,» [En línea]. Disponible: <http://www.innovaportal.com/>. [Último acceso: 11 12 2014].
- [3] LINS, «Nueva Versión del Repertorio de Políticas Sociales,» Montevideo, 2012.
- [4] «Enterprise Application,» [En línea]. Disponible: [http://www.webopedia.com/TERM/E/enterprise\\_application.html](http://www.webopedia.com/TERM/E/enterprise_application.html). [Último acceso: 12 12 2014].
- [5] «What is enterprise software, exactly?,» [En línea]. Disponible: <http://programmers.stackexchange.com/questions/14141/what-is-enterprise-software-exactly>. [Último acceso: 12 12 2014].
- [6] «Introduction to Enterprise Software,» [En línea]. Disponible: <http://www.pearsonhighered.com/samplechapter/0201738295.pdf>. [Último acceso: 12 12 2014].
- [7] «Java EE Platform Overview,» [En línea]. Disponible: <https://docs.oracle.com/cd/E19316-01/820-4342/abfar/index.html>. [Último acceso: 12 12 2014].
- [8] «What is Java-EE Server and different types of Java EE Containers used in Java application Server,» [En línea]. Disponible: <http://javawebaction.blogspot.com/2014/04/Java-EE-Servers.html>. [Último acceso: 12 12 2014].
- [9] «Función de servidor de aplicaciones,» [En línea]. Disponible: <http://msdn.microsoft.com/es-es/library/cc728199%28v=ws.10%29.aspx>. [Último acceso: 29 Noviembre 2014].
- [10] «Java Server Faces Technology,» [En línea]. Disponible: <http://www.oracle.com/technetwork/java/javae/jaserverfaces-139869.html>. [Último acceso: 1 Diciembre 2014].
- [11] «Introducción a la Tecnología Java Server Faces,» [En línea]. Disponible: [http://programacion.net/articulo/introduccion\\_a\\_la\\_tecnologia\\_jaserver\\_faces\\_233](http://programacion.net/articulo/introduccion_a_la_tecnologia_jaserver_faces_233). [Último acceso: 12 12 2014].
- [12] «PrimeFaces FrameWork,» [En línea]. Disponible: <http://www.primefaces.org/>. [Último acceso: 1 Diciembre 2014].
- [13] «Grafico comparación, Google Trends JSF Frameworks,» [En línea]. Disponible: <http://www.google.com/trends/explore#q=primefaces%2C%20richfaces%2C%20icefaces&cmpt=q>. [Último acceso: 2 Diciembre 2014].
- [14] «Primefaces showcase gmap,» [En línea]. Disponible: <http://www.primefaces.org/showcase/ui/data/gmap/basic.xhtml>. [Último acceso: 24 Junio 2014].
- [15] I. A. A. Fernández, «Taller de Sistemas de Información Geográficos Empresariales,» [En línea].

- Disponible: <https://eva.fing.edu.uy/course/view.php?id=429>. [Último acceso: 13 12 2014].
- [16] P. Rebufello, «Introducción a los SIG,» [En línea]. Disponible: <http://www.fing.edu.uy/inco/cursos/sig/clases/Generalidades160812.pdf>. [Último acceso: 13 12 2014].
- [17] «Servicio Web Rest,» [En línea]. Disponible: <http://www.fic.udc.es/files/asignaturas/39ADOO/Tema3Apartado3.2.pdf>. [Último acceso: 30 Mayo 2014].
- [18] «REST vs. SOAP: How to choose the best Webservice,» [En línea]. Disponible: <http://searchsoa.techtarget.com/tip/REST-vs-SOAP-How-to-choose-the-best-Web-service>. [Último acceso: 30 Mayo 2014].
- [19] «Building RESTful Web Services with JAX-RS,» [En línea]. Disponible: <http://docs.oracle.com/javase/6/tutorial/doc/giepu.html>. [Último acceso: 30 Mayo 2014].
- [20] «What are RESTful Web Services,» [En línea]. Disponible: <https://docs.oracle.com/javase/6/tutorial/doc/gijqy.html>. [Último acceso: 30 Noviembre 2014].
- [21] «Definición de Android,» [En línea]. Disponible: <http://www.techopedia.com/definition/14873/android-os>. [Último acceso: 28 Noviembre 2014].
- [22] «Definición de IOS,» [En línea]. Disponible: <http://conceptodefinicion.de/ios/>. [Último acceso: 28 Noviembre 2014].
- [23] «Programming with Objective-C,» [En línea]. Disponible: <https://developer.apple.com/library/ios/documentation/Cocoa/Conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html>. [Último acceso: 28 Noviembre 2014].
- [24] «Cómo crear sitios web optimizados para smartphones,» [En línea]. Disponible: <https://developers.google.com/webmasters/smartphone-sites/?hl=es>. [Último acceso: 7 Octubre 2014].
- [25] «PrimeFaces Mobile Reloaded,» [En línea]. Disponible: <http://blog.primefaces.org/?p=3063>. [Último acceso: 2 Julio 2014].
- [26] D. Panda, R. Rahman, R. Cuprak y M. Remijan, «EJB 3 in Action,» de *EJB 3 in Action Second Edition*, Shelter Island, MANNING, 2014, p. 185.
- [27] «Resource-Based Access Control,» [En línea]. Disponible: <https://stormpath.com/blog/new-rbac-resource-based-access-control/>. [Último acceso: 22 Mayo 2014].
- [28] «Salted Password Hashing - Doing it Right,» [En línea]. Disponible: <https://crackstation.net/hashing-security.htm>. [Último acceso: 16 abril 2014].
- [29] «Java PBEKeySpec,» [En línea]. Disponible: <http://docs.oracle.com/javase/7/docs/api/javax/crypto/spec/PBEKeySpec.html>. [Último acceso: 22 Mayo 2014].
- [30] «Java Secure Random,» [En línea]. Disponible: <http://docs.oracle.com/javase/6/docs/api/java/security/SecureRandom.html>. [Último acceso: 22

Mayo 2014].

- [31] Authorization-RoleBasedAuthorization. [En línea]. Disponible: <http://shiro.apache.org/authorization.html#Authorization-RoleBasedAuthorization>. [Último acceso: 12 Mayo 2014].
- [32] «San Francisco Java User Group presentation on Apache Shiro,» [En línea]. Disponible: <http://shiro.apache.org/2011/01/06/apache-shiro-video-and-slide-from-sf-jug-presentation.html>. [Último acceso: 15 Mayo 2014].
- [33] «Vt-Password,» [En línea]. Disponible: <https://code.google.com/p/vt-middleware/wiki/vtpassword>. [Último acceso: 19 Mayo 2014].
- [34] «Relatórios de Informações Sociais,» [En línea]. Disponible: <http://aplicacoes.mds.gov.br/sagi/RIv3/geral/index.php>. [Último acceso: 10 Junio 2014].
- [35] «Banco Integrado de Programas Sociales, Chile,» [En línea]. Disponible: <http://www.programassociales.cl/>. [Último acceso: 15 Julio 2014].
- [36] «Mapa Interactivo Argentina,» [En línea]. Disponible: <http://www.desarrollosocial.gob.ar/mapa.aspx>. [Último acceso: 27 Agosto 2014].
- [37] «Jsoup,» [En línea]. Disponible: <http://jsoup.org/>. [Último acceso: 15 Agosto 2014].
- [38] «CSS Selectors,» [En línea]. Disponible: [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp). [Último acceso: 15 Agosto 2014].
- [39] «Using Timer Service,» [En línea]. Disponible: <http://docs.oracle.com/javase/6/tutorial/doc/bnboy.html>. [Último acceso: 7 Enero 2014].
- [40] «Develop mobile Java EE applications using Primefaces,» [En línea]. Disponible: <http://www.mastertheboss.com/jboss-web/primefaces/develop-mobile-java-ee-applications-using-primefaces>. [Último acceso: 26 Julio 2014].
- [41] «API Authentictign Realm,» [En línea]. Disponible: [https://shiro.apache.org/static/1.2.3/apidocs/org/apache/shiro/realm/AuthenticatedRealm.html#doGetAuthenticationInfo\(org.apache.shiro.authc.AuthenticationToken\)](https://shiro.apache.org/static/1.2.3/apidocs/org/apache/shiro/realm/AuthenticatedRealm.html#doGetAuthenticationInfo(org.apache.shiro.authc.AuthenticationToken)). [Último acceso: 6 Octubre 2014].
- [42] «Strong passwords hashing with Apache Shiro,» [En línea]. Disponible: <http://blog.swierczynski.net/2013/01/strong-passwords-hashing-wtih-apache-shiro/>. [Último acceso: 27 Febrero 2014].
- [43] «RFC 2616, Protocolo HTTP/1.1,» [En línea]. Disponible: <http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html>. [Último acceso: 17 Julio 2014].
- [44] «Site Up Time,» [En línea]. Disponible: <http://www.siteuptime.com/>. [Último acceso: 20 Noviembre 2014].
- [45] «DotCom Monitor,» [En línea]. Disponible: <https://www.dotcom-monitor.com/>. [Último acceso: 15 Noviembre 2014].

- [46] «Módulo FuzzyStrMatch de PostgreSQL,» [En línea]. Disponible: <http://www.postgresql.org/docs/9.2/static/fuzzystrmatch.html>. [Último acceso: 4 Marzo 2014].
- [47] «PgSimilarity extensión de PostgreSQL,» [En línea]. Disponible: <http://pgsimilarity.projects.pgfoundry.org/>. [Último acceso: 7 Marzo 2014].
- [48] «Balsamic Mockups,» [En línea]. Disponible: <http://balsamiq.com/products/mockups/>. [Último acceso: 30 Octubre 2013].
- [49] «PostgreSQL,» [En línea]. Disponible: <http://www.postgresql.org/>. [Último acceso: 2 diciembre 2014].
- [50] «Java EE at a Glance,» [En línea]. Disponible: <http://www.oracle.com/technetwork/java/javaee/overview/index.html>. [Último acceso: 2 diciembre 2012].
- [51] «Definición de JBoss,» [En línea]. Disponible: <http://whatis.techtarget.com/definition/JBoss>. [Último acceso: 27 Noviembre 2014].
- [52] «What's PostGIS,» [En línea]. Disponible: [http://www.bostongis.com/PrinterFriendly.aspx?content\\_name=postgis\\_tut01](http://www.bostongis.com/PrinterFriendly.aspx?content_name=postgis_tut01). [Último acceso: 1 Diciembre 2014].
- [53] «Feed the Bot,» [En línea]. Disponible: <http://www.feedthebot.com/tools/if-modified/>. [Último acceso: 16 Julio 2014].
- [54] «Definición de Sistemas de Información Geográfica,» [En línea]. Disponible: <http://langleruben.wordpress.com/%C2%BFque-es-un-sig/>. [Último acceso: 29 Noviembre 2014].
- [55] «Apache Shiro,» [En línea]. Disponible: <http://shiro.apache.org/>. [Último acceso: 2 Diciembre 2014].
- [56] «Application Security With Apache Shiro,» [En línea]. Disponible: <http://www.infoq.com/articles/apache-shiro>. [Último acceso: 2 Dicimembre 2014].
- [57] [En línea]. Disponible: <http://steveschols.wordpress.com/>. [Último acceso: 10 10 2014].
- [58] M. I. R. Sosa, «Sistemas de Información Geográfica,» [En línea]. Disponible: <https://eva.fing.edu.uy/course/view.php?id=429>. [Último acceso: 13 12 2014].
- [59] «Java EE 6 CDI,» [En línea]. Disponible: <http://www.mastertheboss.com/jboss-frameworks/cdi/java-ee-6-cdi-example-application>. [Último acceso: 12 12 2014].
- [60] «Openshift,» [En línea]. Disponible: <https://www.openshift.com>. [Último acceso: 01 12 2014].

*LINS - In.Co - Facultad de Ingeniería - UdelaR*

# GUÍA DE RECURSOS DEL MIDES

*Documento de Arquitectura – Proyecto de Grado – Junio  
2014*

**INTEGRANTES:**

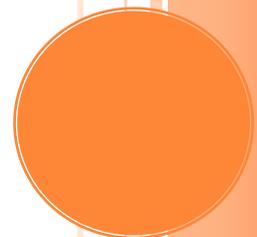
*Maximiliano Arcia*

*Germán Carlosena*

**TUTORAS:**

*Laura Gonzalez*

*Raquel Sosa*





## Índice

1	Introducción.....	3
2	Objetivos .....	3
3	Vista de Casos de Uso .....	4
4	Vista de Componentes.....	6
4.1	HomeBeans.....	7
4.2	Beans.....	7
4.2.1	DetectorNotificadorSingletonBean.....	8
5	Vista de Despliegue .....	9
6	Referencias .....	10

# 1 INTRODUCCIÓN

En este documento se detallará la solución de la nueva versión de la Guía de Recursos del Mides desde el punto de vista arquitectónico, mostrado de forma gráfica con diagramas los componentes y formas de despliegue de la aplicación. Se explicará el fin que cada componente del sistema tiene para la conformación del mismo.

# 2 OBJETIVOS

El objetivo principal de la arquitectura es manejar un modelo que sirva para el desarrollo de la aplicación con los requerimientos funcionales y no funcionales detectados, pero como todo sistema de ingeniería de software necesita otras características que permitan al sistema ser extensible y escalable.

### 3 VISTA DE CASOS DE USO

A continuación se presentan los casos de uso más importantes que hicieron la definición de la arquitectura.

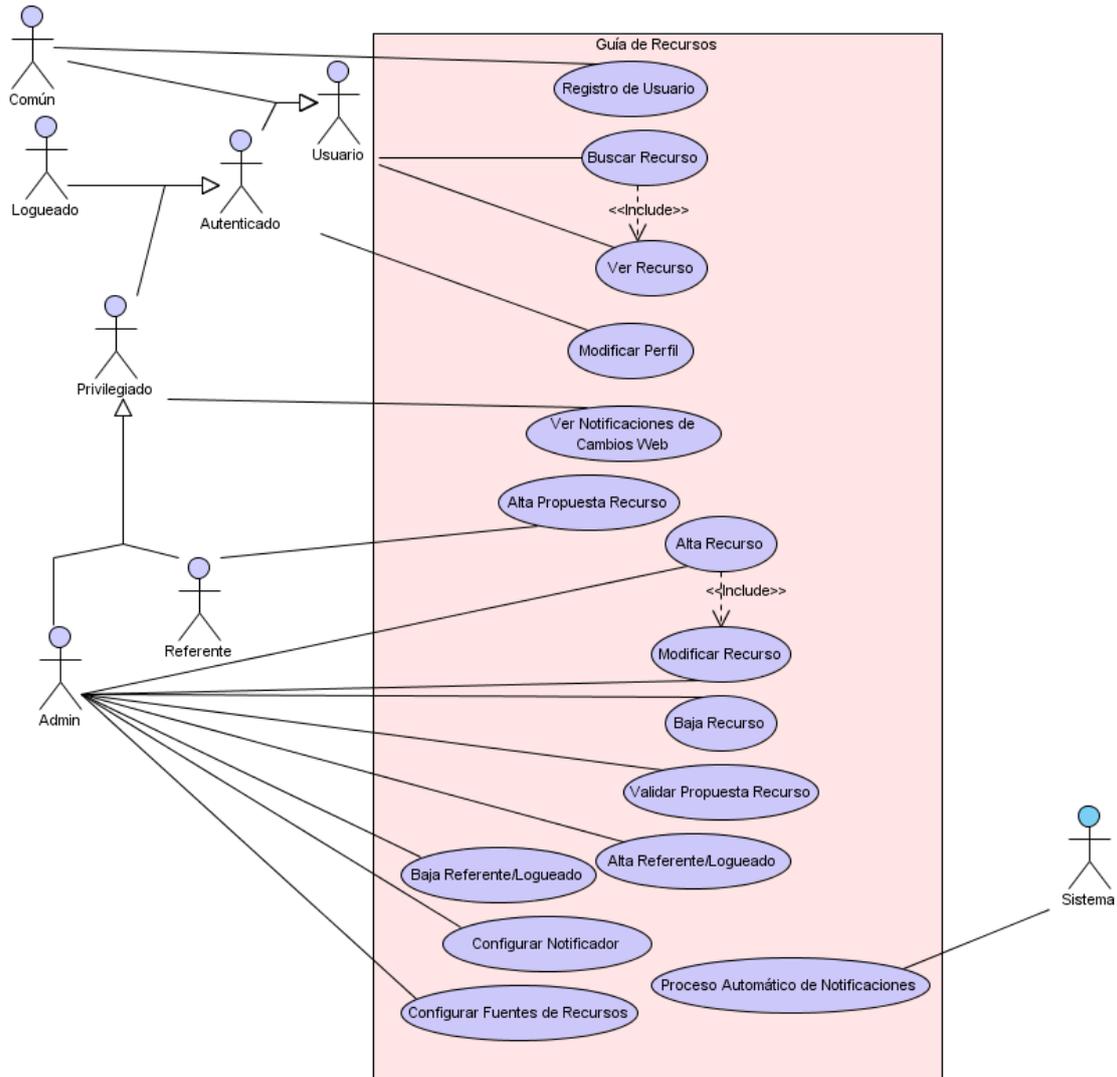
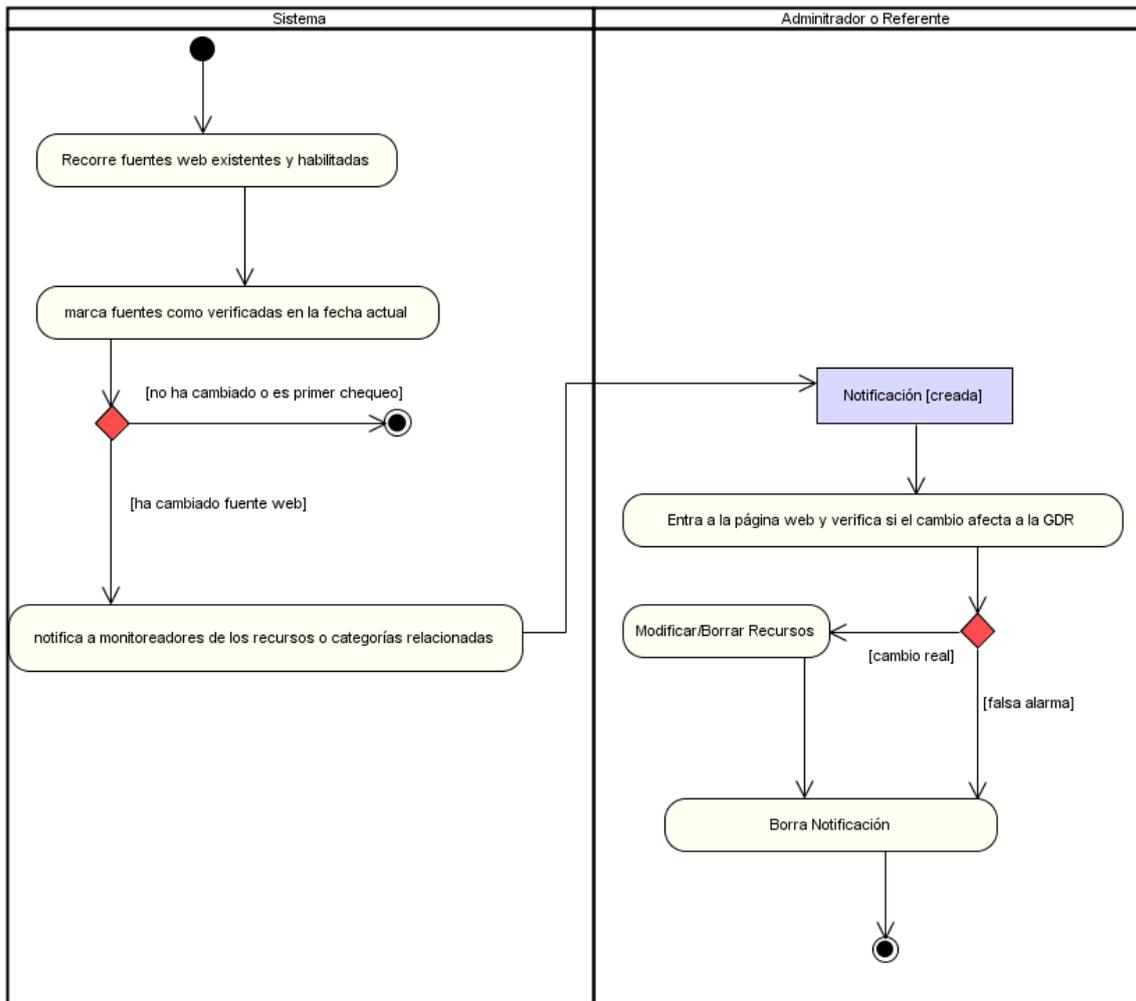


Ilustración 3-1 – Diagrama de Casos de Uso

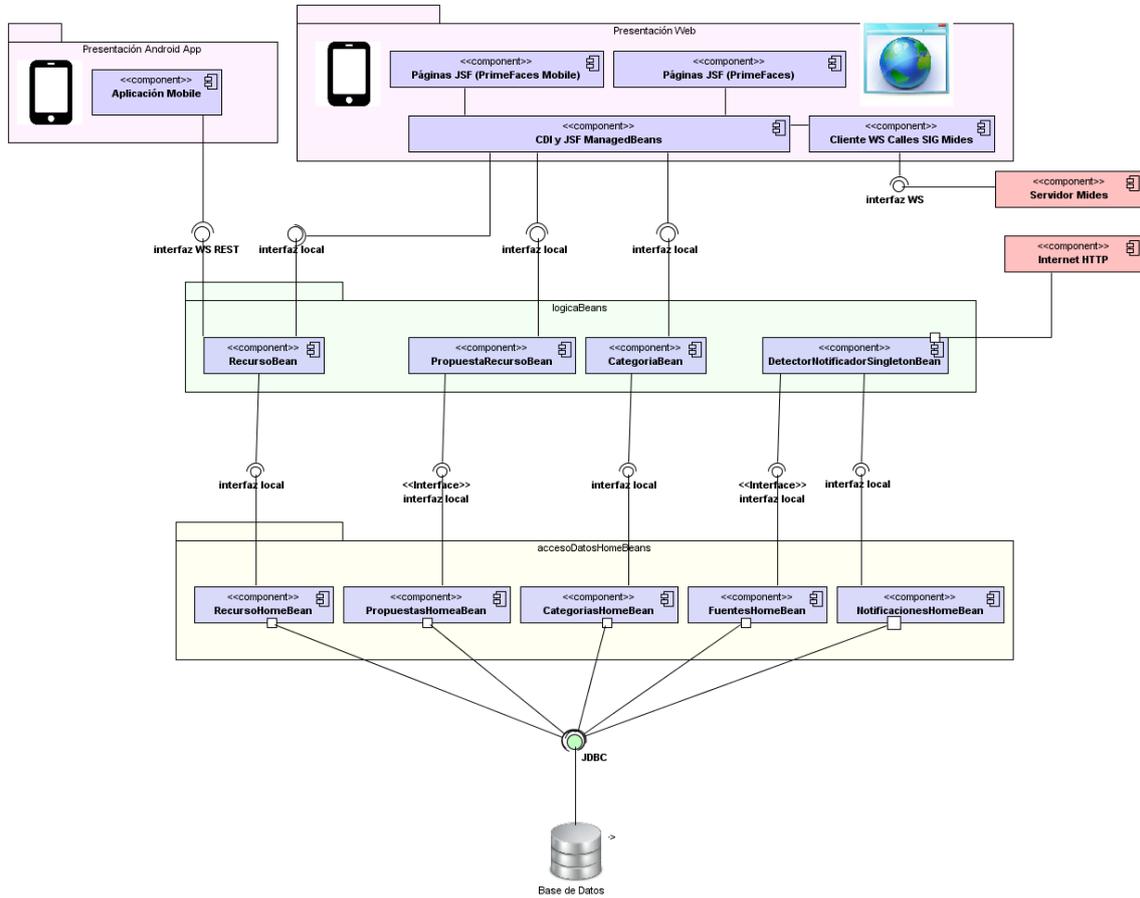
Entre los principales casos de uso se puede destacar el “Proceso Automático de Notificaciones”, el cual se detalla con el siguiente diagrama como es su accionar interactuando con los usuarios del sistema.



**Ilustración 3-2 – Diagrama de Actividad de detector de cambios**

## 4 VISTA DE COMPONENTES

En el siguiente diagrama se muestran los componentes lógicos principales del sistema, cada uno de ellos pueden ser un conjunto de clases, cdi beans, jsf managed beans, ejb beans, páginas jsf, etc, dependiendo del caso. Y entre ellos se conectan con las interfaces detalladas en el diagrama.



**Ilustración 4-1 - Diagrama de componentes de la arquitectura**

A continuación se explicarán los componentes del sistema desde abajo hacia arriba según el diagrama.

## 4.1 HomeBeans

Estos 5 componentes, conforman la capa de acceso a datos:

- RecursoHomeBean
- CategoriaHomeBean
- PropuestaHomeBean
- FuentesHomeBean
- NotificacionesHomeBean

En estos componentes estarán de forma encapsulada todas las consultas, inserciones, modificaciones, y bajas de elementos y asociaciones conectándose hacia la Base de Datos. Cada uno de se encarga de los respectivos elementos el cual su nombre indica: Recursos, Categorías, Fuentes, etc.

Finalmente serán implementados con Stateless Sesion Beans con un Entity Manager inyectado.

## 4.2 Beans

Estos 4 componentes, conforman la capa de acceso a datos:

- RecursoBean
- CategoriaBean
- PropuestaRecursosBean
- DetectorNotificadorSingletonBean

En estos componentes estará desarrollada toda la lógica de negocio, todas las operaciones, altas, bajas, modificaciones y consultas serán realizadas conectándose hacia los HomeBeans, a su vez se harán todos los chequeos necesarios y de no cumplirse alguno de ellos se le comunicara a la capa superior mediante excepciones.

Cada una de los operaciones iniciará una transacción, si llegara a fallar por alguna excepción esperada o inesperada, esta transacción se rollbaqueará automáticamente.

Finalmente serán implementados con Stateless Sesion Beans con un los respectivos HomeBeans inyectados, excptuando el DetectorNotificadorSingletonBean

#### 4.2.1 DetectorNotificadorSingletonBean

Este es un caso particular en la capa lógica por varios motivos: no se comunicará con la capa superior, no será un Stateless Sesion Bean, se iniciará la ejecución automáticamente con una frecuencia configurable y navegará sobre la web para detectar los cambios necesarios. Lo que comparte con los demás beans es que se comunica con la capa inferior para que sus acciones sean guardadas.

Parte de su tarea está especificada en la Ilustración 3-2 – Diagrama de actividad de detector de cambios.

Finalmente será un Singleton Session Bean, con FuentesHomeBean y NotificacionesHomeBean inyectado

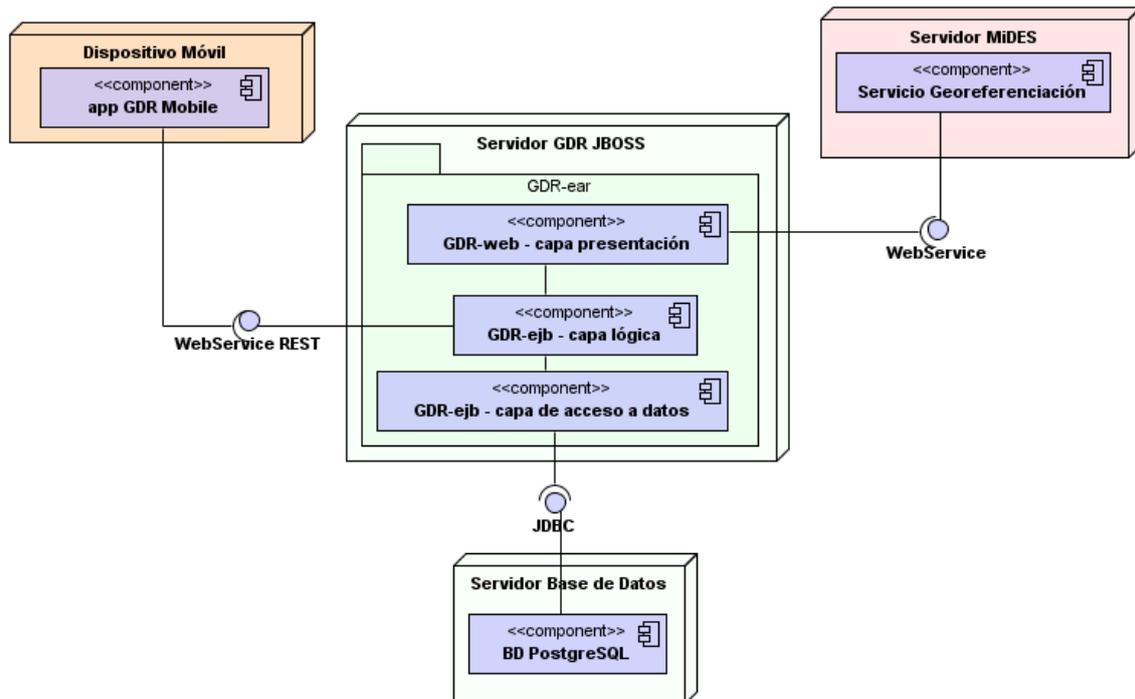
## 5 VISTA DE DESPLIEGUE

Aquí podemos apreciar físicamente la separación de los subsistemas de la GDR en diferentes nodos, esta sería la máxima forma de separación, serían los siguientes nodos:

- Servidor Central GDR JBoss (Jboss 7.1.1)
- Servidor Mides el cual nos ofrece un servicio de georreferenciación
- Servidor de Base de Datos (PostgreSQL 9.2)
- Dispositivo Móvil con aplicación Android

Algo a notar es que no necesariamente todos los subsistemas deben estar separados, la Base de Datos y el Servidor Central podrían ser un mismo servidor físico.

Otra característica destacable es que Jboss podría desplegarse en varios servidores Cluster, mediante configuración y de ser necesario además, algunos cambios de implementación, podría hacerse sin problemas un servicio cluster.



## 6 REFERENCIAS

- <http://www.fing.edu.uy/inco/cursos/tsi/TSI1/2012/teorico/tsi1-1-introduccion.pdf>
- <http://docs.oracle.com/javaee/6/tutorial/doc/bnboy.html>
- <http://docs.oracle.com/javaee/6/tutorial/doc/gipjg.html>
- <http://es.scribd.com/doc/2568098/UML-Diagramas-de-actividad>
- <http://prezi.com/yzvn1klhsd4x/diagramas-de-componentes/>
- [virtual.usalesiana.edu.bo/web/practica/archiv/componen1.ppt](http://virtual.usalesiana.edu.bo/web/practica/archiv/componen1.ppt)
-