

rEvolución: Suite de Web Services para ofrecer nuevas herramientas en la Intranet de Mango

Mario Rodríguez Murillo

Resumen– Dentro de organizaciones profesionales y empresas, las áreas dedicadas a las tecnologías de la información, deben trabajar exhaustivamente en las tareas diarias de elaboración de los productos y desarrollos orientados a sus clientes, optimizando la calidad y el soporte que ofrecen. Sin embargo, para obtener un crecimiento a largo plazo, los equipos IT deben encontrar los procesos e innovaciones digitales que permitan a sus componentes fundamentales, los empleados, trabajar con mejores herramientas y con espacios más adaptados a las singularidades de cada uno. Es por ello que dentro de MANGO, se diseñó un proyecto orientado a mejorar la intranet, el entorno de trabajo común para muchos de los empleados que se encuentran en su central. Un proyecto focalizado en ofrecer nuevas herramientas que faciliten la navegación por este espacio, agilicen tareas diarias y también que lo convierta en un espacio más personal.

Palabras clave– MANGO, intranet, *RESTful*, servicios, *backend*, API, Web, aplicativos

Abstract– Inside professional organizations and companies, the departments dedicated to Information Technology must thoroughly work in diary tasks related to elaborating products and developments orientated to their clients, in order to optimize the quality and support they offer. Nevertheless, in pursuance of obtaining a long-term growth, IT teams must find digital processes and innovations that allow their fundamental components, their workers, to work not only with the best tools but in spaces highly adapted to their own singular needs. This is why MANGO has developed a project directed to improve the intranet, the common work environment used by many of the employees who work in the company central building. This project is focused on offering new tools in order to facilitate navigation through this space, speed up diary tasks and also transform this environment into a more personal setting.

Keywords– MANGO, intranet, *RESTful*, services, *backend*, API, Web, applications



1 INTRODUCCIÓN

REVOLUCIÓN es un proyecto nacido en el seno de MANGO, una multinacional dentro de la industria de la moda dedicada a la creación y comercio de prendas de vestir.

La empresa ha crecido y se ha expandido durante sus más de treinta años de historia, estando actualmente presente en más de cien países y plenamente establecida en el comercio

digital. Este crecimiento ha traído consigo un incremento en el número de empleados que trabajan para la compañía, gran parte de ellos -más de dos mil- se encuentran en las instalaciones principales situadas en Palau-Solità i Plegamans, realizando tareas relacionadas con la gestión de recursos, logística, comunicación y servicios informáticos entre otros.

Para dar soporte a muchas de las actividades y situaciones en las que estas personas participan, MANGO cuenta con su propia intranet[1]. Este lugar supone un punto de encuentro a disposición de los empleados, que como usuarios de la misma, tienen acceso a información, aplicaciones y herramientas disponibles para ayudarles en la ejecución de sus labores diarias.

El proyecto que nos ocupa surge de la necesidad de sacudir la realidad de la actual intranet: han pasado años desde

- E-mail de contacto: m.rodmur@gmail.com
- Mención realizada: Ingeniería del Software
- Trabajo tutorizado por: Yolanda Benítez Fernández (Ciències de la Computació)
- Curso 2016/17

su último proceso de renovación y padece de una estructura poco accesible, de una página principal (de ahora en adelante Home) desaprovechada y de la sensación de que podría ofrecer más recursos beneficiosos a sus usuarios. Para dar solución a esta situación se plantea un procedimiento de transformación, tanto a nivel visual (*frontend*) como de lógica de negocio (*backend*) [2], incluyendo tareas de diseño, desarrollo web y programación de servicios web. Este documento se centra en el trabajo que he realizado: el desarrollo *backend* de una serie de nuevos aplicativos pensados para formar parte de la renovada versión de la Home de la intranet.

2 SITUACIÓN ACTUAL

La situación actual de la Home, espacio que cobijará los elementos enlazados a los servicios web a desarrollar en este proyecto, cuenta con componentes de mucha ayuda para los usuarios, pero también refleja muchas de las carencias que ya se han apuntado en la introducción.

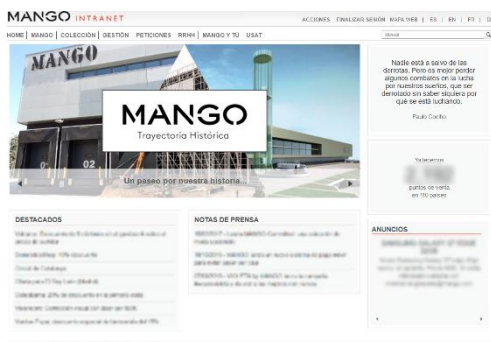


Fig. 1: Versión actual de la Home de la Intranet.

Algunos de sus puntos fuertes a coger como referencia:

- Actúa como acceso principal a un ingente recopilatorio de útiles aplicativos a disposición de los compañeros.
- Posee diferentes métodos para dar soporte al usuario y recoger información de errores y funcionamientos inadecuados.

Por otro lado, los principales aspectos a mejorar son:

- La navegación, principalmente ejercida a través de los accesos que se muestran en la cabecera, se aleja del concepto de diseño orientado a las personas. Por ejemplo, para acceder al tercer nivel de páginas hay que hacerlo desde la propia página de su nivel padre, sin posibilidad de hacerlo desde esta cabecera.
- Los elementos en el área central de la Home, donde un banner corporativo ocupa más de la mitad del espacio, aspecto crítico para quienes trabajan con equipos de pequeña resolución, ya que esta versión no es *responsive*.
- El buscador, herramienta que a partir de una cadena de texto arroja diferentes resultados, desde páginas de la misma intranet, hasta documentos o vídeos, con el problema añadido de que en muchas ocasiones los resultados no tienen relación alguna con la búsqueda.

- El contenido de los elementos del área central, principalmente tarjetas de interacción monodireccional (únicamente del sitio web hacia el usuario) que son en su mayoría notas informativas.

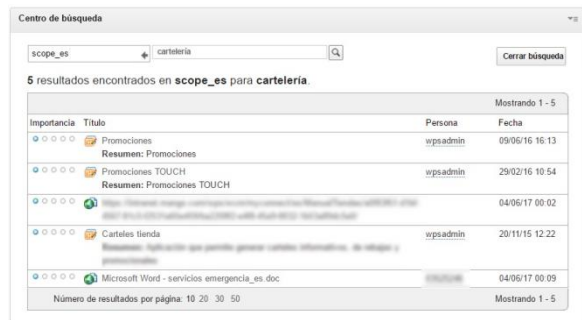


Fig. 2: Ejemplo de operación con el buscador.

Son estos dos últimos puntos, el buscador y las "tarjetas" del apartado central, los que concentrarán los esfuerzos de este trabajo. El diseño de soluciones técnicas, su desarrollo y los planes de pruebas para optimizar su calidad, serán tareas que llevaré a cabo personalmente, mientras que su integración en el portal la ejecutará uno de mis tutores y responsables del proyecto, con la colaboración que yo pueda aportar.

3 OBJETIVOS

3.1 Objetivo principal

El objetivo principal de este proyecto es desarrollar una API de servicios web RESTful[3], dirigida a facilitar el trabajo diario de los empleados, e integrarla en el portal junto al desarrollo visual realizado por los compañeros de diseño y desarrollo *frontend*.

Para determinar los pasos a seguir se han generado diseños de la futurible interfaz y en ella se aprecian algunos de los pequeños aplicativos que se añadirán a la Home.

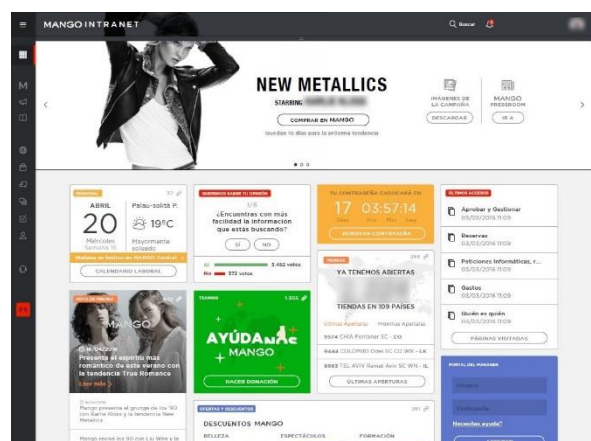


Fig. 3: Una versión del diseño y contenido al que debe aspirar pirar la nueva intranet.

El desarrollo de la *suite* de servicios permitirá disponer de las siguientes aplicaciones en la nueva intranet:

- Últimos accesos/Favoritos: Tarjeta dual que mostrará o bien las últimas páginas a las que ha accedido el usu-

ario, ordenadas de más a menos recientes, o bien las páginas que el usuario decida guardar como favoritas para disponer de un acceso rápido. El usuario podrá añadir nuevas entradas a esos "favoritos" o eliminar alguna de las ya añadidas si no la necesita.

- Expiración de la contraseña: Tarjeta que avisará de la inminente expiración de la contraseña del usuario conectado. Se mostrará una cuenta atrás con el tiempo restante hasta la expiración. Sin embargo, si la cifra que resta es mayor a 15 días, la tarjeta no aparecerá.
- Meteorología: Tarjeta que mostrará información climatológica actualizada para la región desde la que se haya conectado el usuario. Además de esta funcionalidad, debe tener un acceso a la página donde se encuentra el calendario laboral.
- Aprobaciones: Tarjeta que aparecerá únicamente en el caso de que el usuario conectado pertenezca al grupo de aprobadores y tenga aprobaciones pendientes. Aquí se deberán mostrar las aprobaciones pendientes divididas en los grupos a las que pertenecen: generales, facturas, anticipos y agencia de viajes, así como el número total de peticiones pendientes de cada grupo. Se debe incluir en la tarjeta la funcionalidad para aprobar la petición pendiente.
- Encuestas: Tarjeta que mostrará una batería de encuestas para que los usuarios puedan responder y recoger así sus opiniones con el objetivo de seguir mejorando la intranet.
- Tiendas: Otra tarjeta dual que según la elección del usuario mostrará las últimas aperturas o bien las tiendas que tienen su fecha de inauguración más cercana.
- Buscador: Aplicativo que permitirá a los usuarios buscar páginas y/o usuarios a partir de una cadena de búsqueda introducida.

3.2 Objetivos específicos

Para complementar al objetivo principal y garantizar unos niveles de calidad, integridad y adaptación al cambio, se detallan en este apartado una serie de objetivos específicos:

- Objetivo 1: Asegurar la independencia del paquete de servicios, blindando la sostenibilidad futura frente a cambios en el portal o en el diseño.
- Objetivo 2: Aplicar un formato para las peticiones-respuestas en la comunicación de los servicios con la parte "cliente", que permita una integración completa con la parte visual.
- Objetivo 3: Asegurar que el lote de servicios es compatible con los principales dispositivos y navegadores que utilizan los empleados.
- Objetivo 4: Recolectar con regularidad las opiniones de los usuarios respecto a las mejoras y cambios implementados.

- Objetivo 5: Desarrollar un software con una estructura clara y específica, que permita a otros compañeros trabajar en futuras mejoras tras la finalización del proyecto.
- Objetivo 6: Desarrollar un servicio web que almacene información relacionada con la interacción del usuario al navegar por la intranet.
- Objetivo 7: Realizar una búsqueda para obtener una API gratuita que ofrezca información meteorológica actualizada, compatible con las poblaciones donde se encuentran las principales sedes de MANGO.
- Objetivo 8: Adaptar e incorporar al proyecto, el código ya existente que recupera información relacionada con las aprobaciones pendientes de un usuario.

Estos objetivos se han priorizado según la importancia que supone su consecución dentro del alcance del proyecto:

TABLA 1: PRIORIZACIÓN OBJETIVOS ESPECÍFICOS

Identificador	Nivel de prioridad
Objetivo 1	Crítico
Objetivo 2	Crítico
Objetivo 3	Crítico
Objetivo 4	Prioritario
Objetivo 5	Prioritario
Objetivo 6	Prioritario
Objetivo 7	Secundario
Objetivo 8	Secundario

4 METODOLOGÍA

El proyecto rEvolución se ha desarrollado siguiendo la línea del resto de nuevos desarrollos dentro de la organización, utilizando la metodología ágil SCRUM[4].

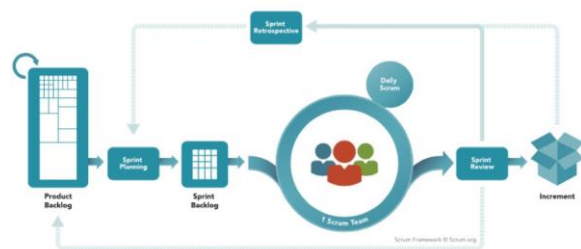


Fig. 4: Ciclo de trabajo en SCRUM.

El equipo ha trabajado siguiendo las actividades propias de la metodología:

- Sesiones de Sprint Plan al inicio del ciclo.
- Reuniones de Daily Plan cada mañana.
- Eventos de Sprint Review al final de cada sprint para evaluar y mostrar el trabajo realizado.
- Jornadas de Sprint Retrospective para valorar el funcionamiento del equipo y consensuar acciones para mejorar como grupo.

Debido al carácter multi-disciplinario del equipo designado para llevar a buen puerto el proyecto de rEvolución, supone una ventaja el hecho de que SCRUM permite un refinamiento para que, a través del *feedback* y de la organización entre los diferentes miembros del equipo, se consigan mejorar y optimizar aspectos del proyecto. La posibilidad que ofrece esta metodología, de sustituir tareas que se encuentren dentro del sprint por otras que aparezcan nuevas con prioridades muy altas, nos ha permitido poder actuar rápidamente ante imprevistos.

5 PLANIFICACIÓN

Inicialmente el proyecto se dividió en fases que pudiesen reflejar entregables de calidad y puntos de control que evaluaran el acierto en términos de planificación.

TABLA 2: FASES DEL PROYECTO

Identificador	Detalle
Fase 1	Preparación del contenido e información de las tareas.
Fase 2	Configuración de los diferentes entornos y programas de trabajo.
Fase 3	Desarrollo de servicios de Últimos accesos y favoritos.
Fase 4	Desarrollo de servicios de Expiración de contraseñas.
Fase 5	Desarrollo de servicios de Aprobaciones pendientes.
Fase 6	Desarrollo de servicios de Tiendas.
Fase 7	Desarrollo de servicios de Meteorología.
Fase 8	Desarrollo de servicios de Encuestas.
Fase 9	Publicación abierta del paquete inicial de servicios.
Fase 10	Desarrollo de servicios de Buscador.

A causa de aplicar SCRUM, su ciclo y su proceso de refinamiento, las fases de la Tabla 2 se adaptaron a sprints de dos semanas de duración (a excepción del Sprint 0, dedicado a la confección del *backlog*).

Las primeras actividades están dirigidas a la configuración de los distintos entornos así como a realizar pruebas y pequeños desarrollos sobre un proyecto ya existente, con el objetivo de familiarizarme con los distintos componentes.

En los sprints posteriores, dedicados a la creación de los distintos aplicativos, las tareas están centradas en:

- **Planificación y análisis:** Los líderes de este proyecto me transmitieron las ideas y restricciones principales que debían seguir y respetar las aplicaciones a crear. Sin embargo, el análisis en detalle de los requisitos más específicos era una tarea que debía realizar por mí mismo bajo su tutela, ya

que la ratificación de este análisis, se aplicaba tras su aprobación. Si mi interpretación se alejaba de una solución óptima, me guiaban hasta el resultado que se adecuaba a los intereses del desarrollo.

- **Implementación y pruebas:** Las actividades de programación que seguían a los análisis fueron ejecutados con un procedimiento más autónomo. A partir de los requisitos y restricciones consensuados, investigaba con el objetivo de implementar el código en las mejores condiciones posibles. Durante y tras el desarrollo, realicé tanto pruebas funcionales, como otras relacionadas con la compatibilidad o la adecuación en términos de eficiencia u optimización.
- **Documentación:** Con el fin de facilitar el mantenimiento de este proyecto así como allanar el camino a futuras mejoras, creé documentación del desarrollo en *Confluence*, la herramienta que usamos para organizar el trabajo de forma escrita.
- **Integración en el portal:** Con la colaboración del equipo de Sistemas, se incluía el código en los servidores de integración, para poder llamar a los servicios web y evaluar el funcionamiento junto a los responsables del proyecto. Haciendo uso de las herramientas de administración del portal e incorporando las llamadas pertinentes, forzábamos la aparición de los diferentes comportamientos que ofrecen estos servicios.

Además de estas actividades, algunos módulos tienen singularidades, como es el caso del módulo de aprobaciones pendientes, para el cual primero hubo que ejecutar una actividad de investigación para obtener una API dedicada a ofrecer información meteorológica.

Otras tareas independientes son las asociadas a preparar el entorno de producción de la intranet de MANGO, con el fin de realizar una primera *release* del proyecto, disponible para el departamento de informática.

En la sección A.1 del apéndice se muestra el desglose de las actividades en detalle.

TABLA 3: SPRINTS DEL PROYECTO

Identificador	Detalle
Sprint 0	Preparación del backlog.
Sprint 1	Familiarización.
Sprint 2	Últimos accesos/Favoritos.
Sprint 3	Últimos accesos/Favoritos (2).
Sprint 4	Expiración contraseña.
Sprint 5	Aprobaciones pendientes.
Sprint 6	Tiendas.
Sprint 7	Meteorología.
Sprint 8	Meteorología (2) y plan de release 1.
Sprint 9	Encuestas.
Sprint 10	Encuestas (2) y ejecución release 1.
Sprint 11	Buscador.
Sprint 12	Buscador (2) + próximos pasos.

6 REQUISITOS

Con el objetivo de detallar las restricciones que debía satisfacer el sistema se elaboró un documento de análisis en el que se describieron cada uno de los requisitos, evaluando tanto su nivel de prioridad como el riesgo que suponía su incumplimiento:

- **Requisitos funcionales:** Comportamiento que debe ofrecer el sistema ante las diferentes peticiones que pone a disposición de los usuarios, teniendo en cuenta sus restricciones, como por ejemplo el tipo de usuario que realiza la petición.
- **Requisitos de usabilidad:** Dirigidos a aspectos como la internacionalización de los aplicativos o los tiempos de ejecución.
- **Requisitos de fiabilidad:** Orientados a la gestión de versiones del Software y comunicación de errores.
- **Requisitos de soporte:** Dedicados a la compatibilidad con dispositivos y navegadores.

En la sección A.2 del Apéndice se encuentra un extracto de los requerimientos especificados.

7 DISEÑO

Un punto clave en la creación de este lote de pequeños aplicativos fue plantearse la estructura que se debía implementar. La forma de gestionar las peticiones a los servicios web y el flujo que debía generar la lógica a aplicar; de manera que las respuestas fuesen óptimas y respetasen las restricciones asociadas a los objetivos y requisitos de este proyecto. Una vez ejecutada la lógica del programa, las respuestas se envían al solicitante haciendo uso de la notación JSON; ya esté compuesta por objetos, estructuras de datos o mensajes de error si es el caso. Por lo tanto, será la parte *frontend* la encargada de interpretar la respuesta y mostrarla al usuario a través de la interfaz web.

La solución pasó por decidir implementar la estructura que se muestra en la Fig.5. El paquete de servicios a modo de bloque externo al portal (lugar desde el cual se generaran las peticiones HTTP mediante javascript) y también independiente a otros sistemas de información que participan en el circuito:

- La base de datos de Oracle que contiene las tablas y atributos que pertenecen al proyecto.
- El servicio de directorios ordenado, accesible a través del protocolo LDAP[5].
- API OpenWeather, una plataforma que bajo ciertas restricciones ofrece un servicio gratuito de información meteorológica bajo solicitudes HTTP.

Con la arquitectura y los procesos de petición-respuesta cubiertos, se planteaba como necesario establecer una estructura común para el código de los diferentes servicios. Maximizar la facilidad de comprensión y mantenimiento y la posibilidad de nuevos desarrollos dentro del alcance de este proyecto eran objetivos destacados. Por ello, se decidió establecer un flujo inspirado en el patrón MVC.

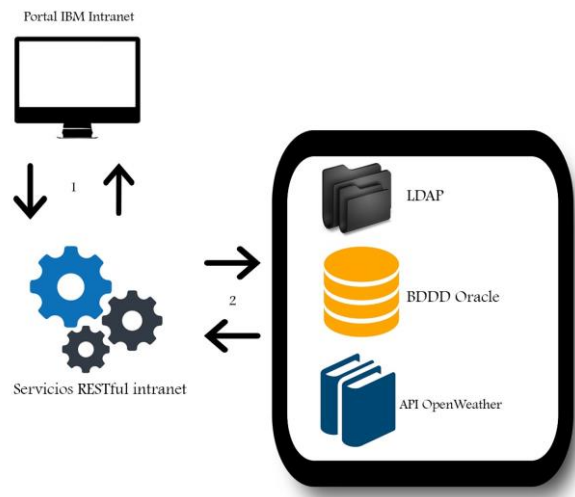


Fig. 5: Estructura de implementación del sistema.

- **Resources:** Los métodos de este paquete representan cada una de las funcionalidades que contienen los distintos módulos. Es en este punto donde se interpreta la petición del portal, evaluando los parámetros que llegan a través de la petición. Estos parámetros serán usados posteriormente en la lógica del proceso, a excepción de un atributo que debe llegar con cada una de las solicitudes sin excepción: un *token* que corresponde al valor asociado a la sesión de la intranet que tiene abierta el usuario en cuestión. Desde estos métodos se encauza el flujo hacia el bloque de *services*.
- **Services:** En este paquete se ejecutarán las operaciones aritméticas o lógicas que no supongan una conexión a base de datos. Además de estas actividades será aquí donde se realicen las peticiones HTTP a otras API, solicitudes a través de LDAP. Este bloque es uno de los más destacable de la estructura, ya que es el centro de muchos esfuerzos. Los distintos *services* convocados tanto desde el paquete de *resources* como desde el de *tests* realizarán la llamada para ejecutar el código que contienen los paquetes de *DAO* y *populator*.
- **DAO (Data Access Object):** Ejecutarán todas las operaciones a realizar con base de datos. En primer lugar se creará una instancia del esquema de BBDD con el que trabajar, para luego realizar la consulta, inserción, eliminación, modificación o llamada a un procedimiento (o un conjunto de estas operaciones), según corresponda.
- **Models:** Las clases de tipo Model representarán a las tablas de las bases de datos y los atributos que estas contienen. La correspondencia se producirá haciendo uso de la librería de persistencia de java. Otro de los propósitos de esta porción del código es alojar las consultas que se lanzarán contra los esquemas de la base de datos.
- **Populators:** La finalidad de estos elementos será formar los objetos *bean* a partir de la información que se ha recuperado a través de los modelos de la base de datos.

- *Beans*: Aquí se encontrarán los objetos que en el último paso del flujo se enviarán haciendo uso de la notación JSON. Los atributos de estos objetos y sus valores serán las respuestas a las peticiones.
- *Tests*: Las pruebas unitarias se efectuarán a través del uso de este paquete. Contendrá los métodos que pondrán a prueba el rendimiento de los servicios.

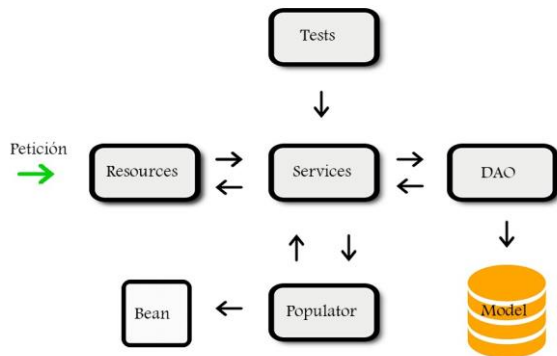


Fig. 6: Ciclo de trabajo en SCRUM

El último paso del diseño tiene como objetivo generar los diagramas de clase de las diferentes aplicaciones que se van a crear en este trabajo. Esta documentación está dirigida a describir los rasgos globales de cada sistema, utilizando nombres de clase que sean sustancialmente auto-explicativos; identificar y plasmar las relaciones entre las diferentes clases que componen el módulo; y mostrar los atributos y métodos que incluyen estas clases.

En el apéndice, en el punto A.3, se muestra el diagrama de encuestas como ejemplo de este material.

8 IMPLEMENTACIÓN Y PRUEBAS

La programación de las de las diferentes piezas destinadas a formar el compendio de estos aplicativos ha sido la actividad de mayor duración. También ha sido este ámbito, en el que el desarrollo de este TFG me ha permitido mejorar más y obtener nuevos y significativos conocimientos. De la misma forma, puesto que los aplicativos han sido desarrollados uno tras otro, siguiendo un orden determinado, si realizamos una comparación entre la implementación del primero y el último de los aplicativos, el progreso es apreciable.

El hecho de haber llevado a cabo este trabajo siguiendo la metodología SCRUM, ha permitido un refinamiento tanto en la codificación como en otros conceptos de la programación a través del ritmo de trabajo aplicado y de las sesiones de revisión y retrospectiva durante los diferentes sprints.

Para ilustrar la manera en la que se ha llevado a cabo este desarrollo hablaré tanto de los lenguajes, como del software utilizado para poder implementar el código.

8.1 Principales lenguajes

Para dar vida al proyecto y aplicar el diseño que se ha mostrado en el apartado anterior, la programación se ha sustentado en dos pilares:

- *Java*: La lógica de los programas desarrollados se ha realizado con Java, el lenguaje orientado a objetos con el que se realizan los distintos aplicativos alojados dentro de la intranet de MANGO. Su esencia permite la programación multiplataforma y pone al servicio del desarrollador una gran cantidad de librerías y frameworks que facilitan el trabajo y permiten obtener un mejor resultado.
- *SQL/PLSQL*: Gran parte de las aplicaciones necesitan de la interacción con la base de datos para responder de forma adecuada, así que la creación de las tablas en la base de datos y las consultas que se ha necesitado incorporar al código se ha hecho en SQL, mientras que los procedimientos se han programado haciendo uso del lenguaje PLSQL de oracle.

Es necesario destacar también el uso de otras notaciones y *frameworks/librerías* que en un segundo plano también han jugado un papel destacado.

- *JSON*: Se ha hecho uso de esta notación sencilla y de fácil interpretación para intercambiar información con la parte javascript que realiza las peticiones a los servicios.
- *Gson*: Librería open-source de google, que nos ha permitido obtener objetos en notación JSON a partir de los objetos Java que obteníamos tras ejecutar toda la lógica de los programas.
- *Spring*: Este reconocido framework ha permitido crear enlaces entre los distintos componentes del software desarrollado. De forma dinámica estas piezas conectan y ensamblan las dependencias en el código, haciendo que los procesos se ejecuten correctamente.
- *javax.persistence*: Clave en el mapeo de los objetos alojados en base de datos y en su representación como modelos, esta librería dedicada a la persistencia de datos, ha permitido el uso de *queries* y transacciones, y de forma más general, poder manipular esos modelos de datos para ofrecer los resultados correctos.

8.2 Principales herramientas

Los principales entornos utilizados han sido los incorporados habitualmente por MANGO en sus desarrollos de Java. A la par con los lenguajes, este programario ha hecho posible el trabajo que se describe en este documento:

- *Eclipse*: Esta conocida herramienta ha permitido la programación del código con su editor, el control de versiones, la interacción con base de datos a través de sus herramientas de configuración y conexión, la compilación y exportación del proyecto y también, la configuración de un servidor Tomcat de Apache para la ejecución del proyecto en un entorno local.
- *PLSQL Developer*: A través de este entorno de desarrollo Oracle, se ha podido acceder a los esquemas y tablas creadas con anterioridad al

proyecto y que se debían utilizar durante el proyecto. Además, la incorporación de nuevas tablas, permisos y *procedures* que se han creado como parte del trabajo también han sido llevados a cabo con esta herramienta.

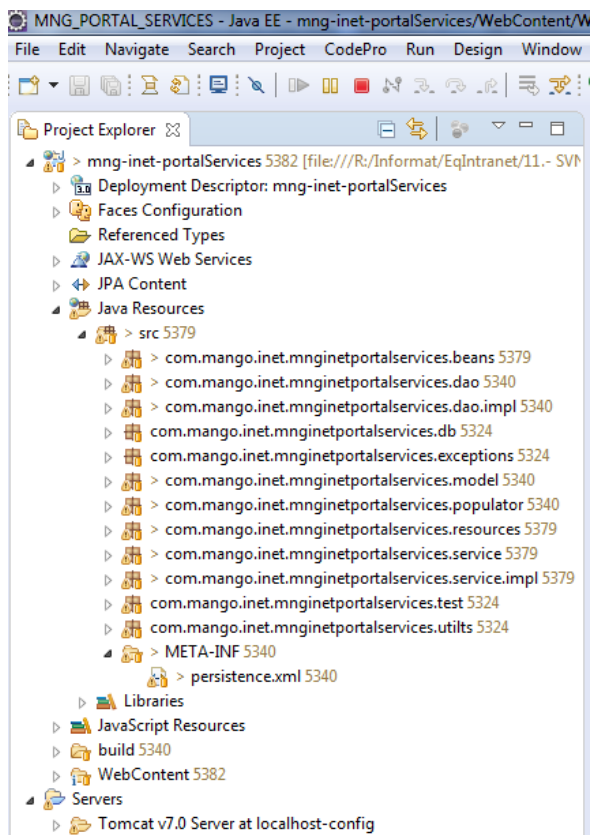


Fig. 7: Estructura del proyecto y los paquetes de código visualizados en Eclipse.

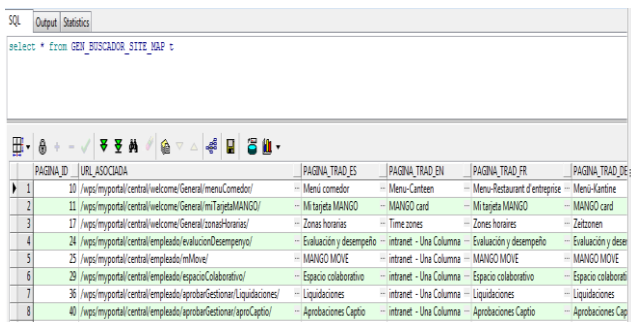


Fig. 8: Consulta a los datos de la tabla de las páginas del buscador a través del PLSQL Developer.

8.3 Plan de pruebas

Este proyecto, como ya se ha descrito en la sección de introducción, está dirigido a un gran número de usuarios (concretamente, los más de 2000 empleados que trabajan en central). Con este apartado de pruebas, el objetivo era realizar un plan de calidad que acompañase al desarrollo, con el objetivo de ofrecer a estos usuarios un producto pulido y con garantías. Para llegar a este fin, se han perseguido los siguientes aspectos:

- Descubrir errores en las aplicaciones antes de liberar

los distintos aplicativos a los usuarios. Cuanto antes sean descubiertos estos errores mejor.

- Minimizar los riesgos y evitar al máximo situaciones con un funcionamiento incorrecto o no deseado.
- Ofrecer un "producto" que cumpla con los requisitos especificados en la fase de análisis de este mismo proyecto.
- Garantizar que la actividad realizada por los distintos aplicativos cumple con la calidad esperada.

Para alcanzar estas metas, se trabajó aplicando una serie de actuaciones:

- Realización de pruebas conjuntas con los responsables del proyecto.
- Creación de test unitarios utilizando la herramienta JUnit.
- *Releases* limitadas a un grupo reducido de usuarios para que prueben las diferentes aplicaciones y reporten los errores o funcionamientos incorrectos que puedan encontrar.
- Plan de pruebas definiendo las funcionalidades que se van a evaluar, el comportamiento esperado que deberían producir y, por último, el resultado real que arrojan al ser ejecutadas.

En la siguiente imagen se puede apreciar el formato del plan de pruebas que se ha aplicado en este proyecto.

T.01.02	Últimos Accesos-Favoritos	Mostrar últimos accesos usuario.	- Existe el empleado. - El empleado ha visitado 9 páginas distintas o menos.	Se devuelve una lista con todas (vacía si solo ha visitado la home) las páginas que el usuario ha visitado, sin repeticiones y ordenadas por fecha.
		Mostrar últimos accesos usuario.	- Existe el empleado. - El empleado ha visitado más de 9 páginas.	Se devuelve una lista las últimas X páginas que el usuario ha visitado (sin repeticiones) ordenadas por fecha.
		Mostrar últimos accesos usuario.	- No existe el empleado.	Mostrar un JSON de error.
T.01.03	Últimos Accesos-Favoritos	Añadir un último acceso a favorito.	-La página debe ser una de las mostradas en los últimos accesos. -La página no era ya favorita.	La tabla de favoritos tiene una nueva entrada que corresponde al usuario y a la página.
T.01.04	Últimos Accesos-Favoritos	Mostrar favoritos usuario.	-Existe el empleado.	Se devuelve una lista con todas (vacía si no tiene favoritos) las páginas que el usuario ha marcado como favoritas, ordenada alfabéticamente.

Fig. 9: Extracto del plan de pruebas que ha sido aplicado.

9 RESULTADOS

9.1 Estado de los objetivos

El objetivo principal de este proyecto era desarrollar una API de *RESTful Web Services* de utilidad para los usuarios de la intranet y convertir la Home en un espacio más

atractivo. Todos los aplicativos que formaban parte del plan del proyecto se han desarrollado y están integrados con la parte *frontend*, de manera que están completamente disponibles para ser llamados desde el portal. Cinco de los siete módulos han sido desarrollados, cumpliendo con todas las restricciones y requisitos. Sin embargo, dos de ellos no han quedado cerrados, los cuales son el módulo del buscador y el de meteorología. En el caso del primero, por falta de tiempo se decidió dejar la posibilidad de buscar personas para más adelante; un contratiempo menor puesto que la búsqueda de páginas, más compleja y prioritaria se ha desarrollado dentro de la planificación. Por lo que respecta a la meteorología, el aplicativo tiene fallos puntuales que lastran la experiencia del usuario, mostrando información que no se corresponde con la realidad. Más adelante, se realizará un análisis en mayor profundidad para solucionar estos inconvenientes y que el aplicativo funcione con mayor estabilidad.

TABLA 4: ESTADO FINAL DEL DESARROLLO DE LOS MÓDULOS RECOGIDOS EN EL OBJETIVO PRINCIPAL

Módulo	Nivel de completitud
Últimos accesos/Favoritos	Completado
Expiración contraseña	Completado
Meteorología	Muy avanzado
Aprobaciones	Completado
Encuestas	Completado
Tiendas	Completado
Buscador	Muy avanzado

Por lo que respecta a los objetivos específicos, listados en el apartado 3.2 de este mismo documento, han sido completados todos los objetivos a excepción del objetivo número siete. Este objetivo, relacionado con la inclusión de la API gratuita OpenWeather, que ofrece información meteorológica, se da por completado pero con matices. La API está cumpliendo su función pero no se "oficializará" su consecución hasta determinar la fuente del mal funcionamiento de la aplicación de meteorología desarrollada.

Debido al grado de completitud de los objetivos y a la madurez alcanzada por el proyecto, se da por cumplido el objetivo de ofrecer un compendio de herramientas destinado a los usuarios de la intranet, puntualizando, eso sí, que el desarrollo no ha finalizado y restan aspectos por pulir.

9.2 Resultados del desarrollo

Es momento de mostrar el desarrollo finalizado y ya integrado en el portal de MANGO (actualmente restringido a un colectivo determinado de usuarios, pero con el mismo formato que la versión que se liberará para los empleados de Central). A continuación aparecen las diferentes *cards* que contiene la nueva versión de la Home de la intranet. Estas pequeñas aplicaciones son las que muestran las respuestas que dan los servicios web a las peticiones realizadas por la implementación *frontend* y con las que pueden interactuar los usuarios del sitio. Puntualizar que el árbol de páginas y contenido no se ha adaptado aún a esta nueva versión de la intranet y hay información temporal que devuelve y que será sustituida por la auténtica en el futuro,

cuando la inclusión del contenido se realice.



Fig. 10: Resultado de la tarjeta de la Home que se alimenta del servicio web de últimos accesos/favoritos.



Fig. 11: Resultado de la tarjeta de la Home que se alimenta del servicio web de expiración de la contraseña.



Fig. 12: Resultado de la tarjeta de la Home que se alimenta del servicio web de aprobaciones.



Fig. 13: Resultado de la tarjeta de la Home, que se alimenta del servicio web de meteorología.



Fig. 14: Resultado de la tarjeta de la Home que se alimenta del servicio web de tiendas.



Fig. 15: Resultado de la tarjeta de la Home que se alimenta del servicio web de encuestas.



Fig. 16: Resultado de la herramienta de la Home que se alimenta del servicio web del buscador.

10 CONCLUSIONES

10.1 Evaluación de los requisitos

De los sesenta requisitos especificados en el documento de análisis se han alcanzado con éxito cincuenta y seis de ellos. Los que no han llegado a finalizarse son dos requisitos relacionados con procesar la respuesta de información meteorológica actual y predictiva; el requisito asociado a la utilización del buscador para encontrar a otros empleados y, por último, el requisito relacionado con implementar el contenido adaptado a cuatro idiomas: español, inglés, francés y alemán. La implementación de los dos últimos idiomas ha quedado prorrogada.

El número de requisitos completados se asemeja en gran medida a la cifra de requisitos planteados, acercando el resultado actual al deseado en un inicio.

La ausencia entre los requerimientos completados, del requisito de recuperar información de personas haciendo uso del buscador, se debe en gran medida a la re-planificación provocada por los errores imprevistos producidos en el módulo de meteorología. En un inicio no se contemplaron tantas dificultades para implementar este módulo, que a priori, parecía de los más sencillos y en el refinamiento del desarrollo, se ha priorizado optimizar el funcionamiento de uno de los servicios ya desarrollados por encima de desarrollar una nueva funcionalidad. Es por ello que en las últimas jornadas del proyecto se trabajó en estos errores aplazando ligeramente la implementación del buscador de usuarios.

10.2 Evaluación de las fases

Referente a la planificación, tan sólo una de las fases no se ha llegado a desarrollar. El resto de ellas han sido concluidas, quedando pendientes la implementación y corrección del código referente a los requisitos incompletos del apartado anterior.

TABLA 5: REVISIÓN DE LAS FASES DEL PROYECTO

Fase	Compleitud
Fase 1 - Concluida	Preparación del contenido e información de las tareas
Fase 2 - Concluida	Configuración de los diferentes entornos y programas de trabajo.
Fase 3 - Concluida	Desarrollo de servicios de Últimos accesos y favoritos.
Fase 4 - Concluida	Desarrollo de servicios de Expiración de contraseña.
Fase 5 - Concluida	Desarrollo de servicios de Aprobaciones pendientes.
Fase 6 - Concluida	Desarrollo de servicios de Tiendas.
Fase 7 - Concluida	Desarrollo de servicios de Meteorología.
Fase 8 - Concluida	Desarrollo de servicios de Encuestas.
Fase 9 - No realizada	Publicación abierta del paquete inicial de servicios.
Fase 10 - Concluida	Desarrollo de servicios del Buscador.

Se llegó a planificar y realizar los preparativos para ejecutar una sesión de prueba con diferentes usuarios, previa a la *release* abierta. En ella los usuarios iban a realizar pequeñas actividades sobre la nueva versión de la Home con el objetivo de ver si el sistema se comportaba de forma consecuente y su *feedback* era positivo. Finalmente esta sesión se anuló por motivos ajenos al desarrollo y la publicación quedó aplazada para más adelante, con la incorporación de un pequeño módulo que permita a los usuarios cambiar entre la nueva y la antigua versión a su antojo.

Hubo tareas que duraron algo menos de lo previsto y con otras sucedió lo contrario. La posibilidad de contar con un espacio dedicado a la revisión del sprint permitió realizar ajustes de forma regular en la planificación de las tareas, compensando estos desfases y haciendo que finalmente el resultado se asemeje en gran medida a lo proyectado en la planificación inicial. El contar con estos encuentros que proporciona SCRUM supuso un gran activo a la hora de no desviarse notablemente del camino señalado.

10.3 Estado actual

Con el proyecto finalizado y tras haber analizado las diferentes métricas en forma de objetivos y requisitos, y el resultado que arroja el funcionamiento de los diferentes servicios web implementados, puedo valorar que la motivación de este proyecto se ha cumplido. Con este lote de nuevas herramientas se cubren distintas necesidades

identificadas en la interacción de los empleados con la intranet, como por ejemplo, facilitar la navegación, agilizar trámites diarios o fomentar a mayor escala el sentimiento de colectivo. El hecho de implementar el código como servicios web ha conducido a una fácil separación de los desarrollos *frontend* y *backend*, sin causar perjuicios a la hora de integrar todo el sistema, como también ha permitido que el conjunto de aplicativos funcione en la Home de la intranet mediante llamadas asíncronas, obteniendo buenos tiempos de respuesta que hacen que la navegación sea rápida y agradable. Por último, el uso de la metodología SCRUM y las opiniones y recomendaciones que han hecho los compañeros que han participado como *testers* han facilitado un proceso de continua mejora y refinamiento del desarrollo.

10.4 Líneas futuras

Uno de los propósitos de este proyecto ha sido diseñar e implementar este lote de mejoras, de forma que futuros desarrollos pudieran ser acometidos con facilidad. La estructura empleada facilita que a medida que en el futuro se identifiquen nuevas necesidades, se puedan diseñar soluciones e incluirlas en este paquete de servicios.

A corto plazo, para mejorar el contenido ya existente se debería:

- Resolver los errores registrados.
- Finalizar el desarrollo del buscador con la opción de buscar personas.
- Crear un pequeño *script* que permita al usuario, cambiar entre la nueva y la antigua versión de la Home según su conveniencia.
- Desarrollar aplicativos de mantenimiento para las herramientas de encuestas y buscador. Ambas se alimentan de grandes cantidades de datos; la relación de preguntas y respuestas de las encuestas por un lado y todas las páginas de la intranet con su correspondiente información, por el lado del buscador. De este modo se facilitará y agilizará la creación y gestión de esta información en base de datos, actividad llevada a cabo por perfiles centrados en la administración de sitios web y portales, poco familiarizados con los sistemas gestores de bases de datos.

10.5 Conclusiones personales

Este trabajo ha supuesto para mí, un punto de inflexión dentro del recorrido del grado universitario. Ha sido un proyecto fuera del ámbito académico, atractivo y dinámico, en el que he tenido libertad para aportar mis ideas, crecer y recaptar aprendizaje a en grandes dosis.

He podido participar en un proyecto orientado a una cantidad sustancial de usuarios y mejorar no sólo en cuanto a conocimientos de diseño, programación y pruebas del software se refiere, sino también a procesos de planificación y refinamiento dentro de un equipo multifuncional dentro de las metodologías ágiles.

AGRADECIMIENTOS

Esta experiencia que me llevo no habría sido posible sin la

colaboración de ciertas personas. En primer lugar quiero agradecer el tutelaje y supervisión de mis tutores tanto de la universidad como de la compañía, Yolanda Benítez y Josep Antoni Pérez. El conjunto del proyecto siempre se ha llevado adelante con empuje y eso ha sido posible gracias a la implicación del equipo, mi sincero aprecio y afecto a Rodolf Moncusí, Xavier Verdaguer, Enrique García y Laura Yagüe.

Agradecer la confianza de mis padres y en especial la firme convicción de mi madre, que han hecho posible que pese a los contratiempos, llegase a subir un peldaño más y alcanzase esta etapa.

Por último me gustaría agradecer la disposición constante de mi pareja, Agnès, para acompañarme y ayudarme en todo momento.

Llegar a este punto ha sido posible gracias a la corriente que ha generado la voluntad de muchas otras personas y me ha conducido a este momento. A todas ellas, muchas gracias.

REFERENCIAS

- [1] Jack Schofield, "What are intranets and extranets?", BBC, Septiembre 2010, [En línea]. Disponible en: <http://www.bbc.co.uk/webwise/guides/intranets-and-extranets>. Consulta: 27 de febrero de 2017.
- [2] Pluralsight, "What's the Difference Between the Front-End and Back-End?", Pluralsight, Enero de 2015, [En línea]. Disponible en: <https://www.pluralsight.com/blog/film-games/whats-difference-front-end-back-end>. Consulta: 28 de febrero de 2017.
- [3] Leonard Richardson y Sam Ruby, "Restful Web Services" Web services for the real world, O'Reilly, Mayo 2007. Capítulo 3: "What makes RESTful services different?". Consulta: 27 de febrero de 2017.
- [4] Ken Schwaber, Jeff Sutherland, "The definitive guide to Scrum: The rules of the game", The Scrum Guide, pp. 03-12, Julio 2016 [En línea]. Disponible en: <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf>. Consulta: 3 de marzo de 2017.
- [5] UnboundID, "Basic LDAP concepts", 2015. [En línea]. Disponible en: <https://www.ldap.com/basic-ldap-concepts>. Consulta: 9 de abril de 2017.

A.1 Planificación de Sprints y actividades

	Nombre de tarea	Duración	Predecesoras	Nombres de los recursos
30	↳ Proyecto Revolución	74 días		Mario
31	↳ Sprint 0	2 días		Mario
32	Preparación Tareas Backlog	2 días		
33	↳ Sprint 1 - Familiarización	6 días	31	Mario
34	Instalación y configuración entornos	3 días		
35	Pruebas y familiarización proyecto existente	2,5 días	34	
36	Tareas Scrum	0,5 días	35	
37	↳ Sprint 2 - Últimos accesos/Favoritos	6 días	33	Mario
38	Planificación y Análisis	0,5 días		
39	Implementación y pruebas	4,5 días	38	
40	Documentación	0,5 días	39	
41	Tareas Scrum 2	0,5 días	40	
42	↳ Sprint 3 - Últimos accesos/Favoritos(2)	6 días	37	Mario
43	Implementación y pruebas	4,5 días		
44	Documentación	0,5 días	43	
45	Integración en el portal	0,5 días	44	
46	Tareas Scrum 3	0,5 días	45	

Fig. 17: Primera parte de la planificación.

	Nombre de tarea	Duración	Predecesoras	Nombres de los recursos
47	↳ Sprint 4 -Expiración Contraseña	6 días	42	Mario
48	Investigación control actual contraseña	0,5 días		
49	Planificación y Análisis	0,25 días	48	
50	Implementación y pruebas	4 días	49	
51	Documentación	0,25 días	50	
52	Integración en el portal	0,5 días	51	
53	Tareas Scrum 4	0,5 días	52	
54	↳ Sprint 5 - Aprobaciones Pendientes	6 días	47	Mario
55	Investigación control actual peticiones	0,5 días		
56	Planificación y Análisis	0,25 días	55	
57	Implementación y pruebas	4 días	56	
58	Documentación	0,25 días	57	
59	Integración en el portal	0,5 días	58	
60	Tareas Scrum 5	0,5 días	59	
61	↳ Sprint 6 - Tiendas	6 días	54	Mario
62	Investigación control actual mensaje tiendas	0,5 días		
63	Planificación y Análisis	0,25 días	62	
64	Implementación y pruebas	4 días	63	
65	Documentación	0,25 días	64	
66	Integración en el portal	0,5 días	65	
67	Tareas Scrum 6	0,5 días	66	

Fig. 18: Segunda parte de la planificación.

	Nombre de tarea	Duración	Predecesoras	Nombres de los recursos
68	↳ Sprint 7 - Meteorología	6 días	61	Mario
69	Investigación APIs meteorológicas	1 día		
70	Planificación y Análisis	0,5 días	69	
71	Implementación y pruebas	3,5 días	70	
72	Documentación	0,5 días	71	
73	Tareas Scrum 7	0,5 días	72	
74	↳ Sprint 8 - Meteorología (2) y plan de release 1	6 días	68	Mario
75	Implementación y prueba	1,5 días		
76	Diseño Técnico Encuesta:	2 días	75	
77	Planificación release 1	1,25 días	76	
78	Documentación	0,25 días	77	
79	Integración en el portal	0,5 días	78	
80	Tareas Scrum 8	0,5 días	79	
81	↳ Sprint 9 - Encuestas	6 días	74	Mario
82	Planificación y Análisis	0,5 días		
83	Implementación y pruebas	4,5 días	82	
84	Documentación	0,5 días	83	
85	Tareas Scrum 9	0,5 días	84	

Fig. 19: Tercera parte de la planificación.

	Nombre de tarea	Duración	Predecesoras	Nombres de los recursos
86	↳ Sprint 10 - Encuestas (2) y release 1	6 días	81	Mario
87	Implementación y prueba	3 días		
88	Documentación	0,25 días	87	
89	Integración en el portal	0,5 días	88	
90	Preparación y Ejecución release 1	1,75 días	89	
91	Tareas Scrum 10	0,5 días	90	
92	↳ Sprint 11 - Buscador	6 días	86	Mario
93	Planificación y Análisis	1 día		
94	Implementación y prueba	4,5 días	93	
95	Tareas Scrum 11	0,5 días	94	
96	↳ Sprint 12 - Buscador (2) + próximos pasos	6 días	92	Mario
97	Implementación y prueba	3,5 días		
98	Documentación	0,75 días	97	
99	Integración en el portal	0,75 días	98	
100	Planteamiento próximas mejoras	0,5 días	99	
101	Tareas scrum 12	0,5 días	100	

Fig. 20: Cuarta parte de la planificación.

A.2 Ejemplos del documento de análisis

Identificador del requisito	R1.07.08
Título	Funcionalidades/Buscador/Resultados búsqueda
Descripción	El sistema debe ser capaz de ofrecer como respuesta a la búsqueda de un usuario, un listado de las opciones coincidentes, cada una de estas con el nombre de la página, la descripción y el enlace a la página.
Prioridad	2
Riesgo	M
Referencias	

Identificador del requisito	R1.07.09
Título	Funcionalidades/Buscador/Priorización resultados
Descripción	El sistema debe aplicar una priorización para mostrar el listado de resultados, determinada por el aspecto en el que se produce la coincidencia. De mayor a menor prioridad: -El usuario ya realizó esta búsqueda en el pasado. -Si un número de usuarios > X escogieron la misma opción para la búsqueda que ha realizado el usuario. - Hay coincidencia en el nombre de la página y la cadena de búsqueda. - Hay coincidencia en la descripción de la página y la cadena de búsqueda. - Hay coincidencia en la url de la página y la cadena de búsqueda. - Hay coincidencia en el tag asociado a la página y la cadena de búsqueda. Aparte del orden, se extraerá el valor de la ponderación para conocer si es una entrada específicamente priorizada que debe aparecer al inicio.
Prioridad	1
Riesgo	H
Referencias	

Fig. 21: Algunos de los requisitos funcionales del módulo del buscador.

Identificador del requisito	R1.02.02
Título	Funcionalidades/Expiración contraseña/Condiciones usuarios
Descripción	Los usuarios de los que el sistema recuperará la información referente a la expiración, deben ser empleados de central y pertenecer actualmente a la compañía.
Prioridad	1
Riesgo	C
Referencias	

Identificador del requisito	R1.02.03
Título	Funcionalidades/Expiración contraseña/Obtención aviso expiración
Descripción	El sistema debe proveer de una notificación con el tiempo que aún tiene la contraseña de un usuario de central como válida, pero en este caso sólo si el periodo es inferior a 15 días, en el caso opuesto, no devolverá notificación alguna. El formato del tiempo debe ser dd MM:HH:SS.
Prioridad	1
Riesgo	C
Referencias	

Fig. 22: Algunos de los requisitos funcionales del módulo de expiración de la contraseña.

Identificador del requisito	R2.01.01
Título	Usabilidad/Idiomas/Multi-Idioma
Descripción	El software del sistema, debe estar adaptado a los idiomas con los que permite navegar la Intranet: Español, Inglés, Francés y Alemán.
Prioridad	3
Riesgo	C
Referencias	

Identificador del requisito	R2.02.01
Título	Usabilidad/Velocidad/Tiempo carga home
Descripción	El software del sistema, debe ser eficiente y ofrecer un tiempo de carga para el conjunto de todos los servicios de la home, en torno a 2 segundos sin penalizaciones en uno u otro navegador.
Prioridad	1
Riesgo	C
Referencias	

Fig. 23: Requisitos de usabilidad.

Identificador del requisito	R3.01.01
Título	Fiabilidad/Comunicación de Errores/Envío mail
Descripción	El sistema deberá proveer de un sistema de envío de mails al personal responsable, en casos en los que no se pueda garantizar el correcto funcionamiento del componente el 100% del tiempo. Se hará comunicando el tipo de error que se ha producido, y si este es conocido, la actuación necesaria para corregir la situación manualmente.
Prioridad	1
Riesgo	C
Referencias	

Identificador del requisito	R3.02.01
Título	Fiabilidad/Versiones/Gestión de versiones
Descripción	El sistema deberá tener la configuración adecuada en eclipse para poder ser gestionado mediante el plugin de SVN (SW de control de versiones).
Prioridad	1
Riesgo	C
Referencias	

Fig. 24: Requisitos de fiabilidad.

Identificador del requisito	R4.01.01
Título	Soporte/Compatibilidad/Navegadores
Descripción	El sistema debe ser compatible con los navegadores utilizados por el personal de Mango: Firefox, Chrome e Internet Explorer (especialmente este último)
Prioridad	1
Riesgo	C
Referencias	

Identificador del requisito	R4.01.02
Título	Fiabilidad/Compatibilidad/Dispositivos móviles
Descripción	El sistema deberá ofrecer las mismas posibilidades y el mismo buen funcionamiento en dispositivos móviles.
Prioridad	1
Riesgo	C
Referencias	

Fig. 25: Requisitos de soporte

A.3 Diseño de clases

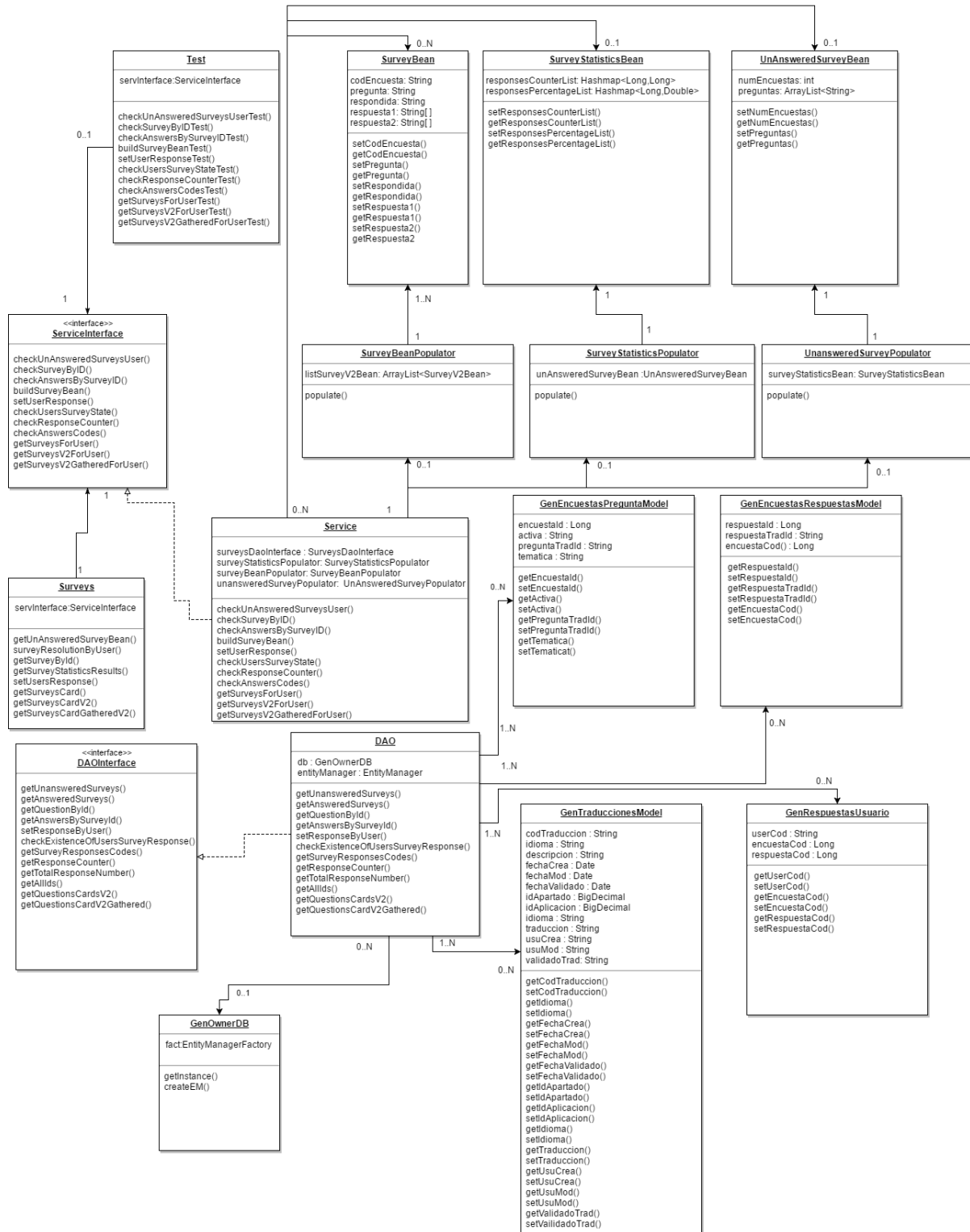


Fig. 26: Diagrama de clases del módulo encuestas