



Universidad de Oviedo



Escuela de
Ingeniería
Informática

GestUsers: Sistema de Gestión de Usuarios



*Arquitectura Software para GestUsers.
Descripción del trabajo práctico (2018)*

Descripción de la práctica primera a realizar por los grupos de trabajo de la asignatura de Arquitectura del Software durante el curso 2017-18.

Escuela de Ingeniería Informática, Univ.
Oviedo

19 de febrero de 2018

GestUsers: Sistema de Gestión de Usuarios

Autores:

Pablo González Martínez - UO245699 – E3B1

Diego Álvarez Guinarte – UO251682 – E3B1

Manuel Junco Díez - UO252010 – E3B1

Iván Suárez Castiñeiras - UO244730 – E3B1

José Antonio Marín Álvarez - UO212006 – E3B2

Miguel Martínez Serrano – UO237030 - E3B2

Daniel Martínez Valeriano – UO252438 – E3B2

Andrés Ángel González Granda – UO68216 - E3B2

Kilian Pérez González – UO21504 – E3B2

Fecha: 19 de febrero de 2018

Versión: 2018.ES.001

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 2 de 37

Tabla de contenido

1	Introducción y Objetivos	5
2	Requisitos	6
2.1	Loader.....	6
2.2	Agents.....	7
3	Metodología usada.....	8
4	Identificación de <i>Stakeholders</i>	9
4.1	Alumnos del grupo E3B1	9
4.2	Alumnos del grupo E3B2	9
4.3	Administrador del Sistema	9
4.4	Agents.....	10
4.5	Profesores de la asignatura.....	10
5	Atributos de calidad	11
5.1	Lista de atributos de calidad	12
5.2	Atributos de calidad ^[OBJ]	13
6	Restricciones	14
6.1	Restricciones técnicas	14
6.2	Restricciones organizativas	14
7	Ámbito del sistema y contexto.....	15
8	Escenarios de calidad	17
9	Vistas	20
9.1	Contexto.....	20
9.1.1	Presentación principal.....	20
9.1.2	Catálogo de elementos	20
9.2	Loader.....	23
9.2.1	Presentación principal.....	23
9.2.2	Catálogo de elementos	24
9.2.3	Diagrama contextual	26
9.2.4	Justificación de las decisiones	27
9.3	Agents.....	29
9.3.1	Presentación principal.....	29
9.3.2	Catálogo de elementos	29
9.3.3	Diagrama contextual	31
9.3.4	Justificación de las decisiones	32
9.4	Vista de Paquetes.....	35

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 3 de 37

9.4.1	Presentación principal.....	35
9.4.2	Catálogo de elementos	35
9.4.3	Diagrama contextual	35
9.4.4	Justificación de las decisiones	35
9.5	Vista de Despliegue	36
9.5.1	Presentación principal.....	36
9.5.2	Catálogo de elementos	36
9.5.3	Diagrama contextual	36
9.5.4	Justificación de las decisiones	36
10	Bibliografía	37

1 Introducción y Objetivos

El objetivo de este documento es definir la estructura de una arquitectura para gestión de usuarios que pueda ser reutilizada. Aunque el sistema aquí descrito tiene funcionalidad propia, el objetivo es que pueda integrarse como parte de un sistema general de participación ciudadana.

La arquitectura aquí descrita forma parte del primer entregable de la asignatura Arquitectura del Software, impartida en el grado de Ingeniería Informática del Software, Escuela de Ingeniería Informática, Universidad de Oviedo por los autores de este documento.

El sistema se ha descompuesto en dos partes: Loader para cargar los datos de los agentes que podrán enviar incidencias al sistema, y Agents, para consultar y obtener información de los agentes que participan en el sistema. Los estudiantes deberán implementar el software descrito en este documento en dos equipos de 3 ó 4 personas durante 3 semanas. Un equipo desarrollará el módulo Loader y otro equipo desarrollará el módulo Agents.

En los próximos entregables, los estudiantes crearán la arquitectura e implementarán un prototipo del resto del sistema de participación ciudadana del cual formará parte el sistema actual.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 5 de 37

2 Requisitos

La gestión de usuarios se hará siguiendo un esquema de dos partes:

- Loader: Carga la lista de agentes que podrán cargar incidencias.
- Agents: Permite consultar y obtener información de los agentes que participan en el sistema.

2.1 Loader

El Administrador del sistema debe poder introducir los datos de los agentes, dichos datos provienen de un Excel que será entregado por la organización. Puede haber distintos tipos de agentes como un "ciudadano", una "entidad", "sensor", etc.

La introducción del padrón y las listas se realizará a partir un formato neutro, en base a ficheros Excel formados por filas de personas, cada una con la siguiente información:

- Nombre
- Localización (coordenadas, opcional para personas y entidades)
- Email
- Identificador
- Tipo

Durante la importación de estos datos se creará un usuario y una clave que le permita acceder al sistema para visualizar la información almacenada.

Si un agente figura en dos listas diferentes, esta eventualidad debe ser identificada, registrada en el fichero de log y el usuario sólo podrá ser creado una vez. Si los datos del usuario no son los mismos, no se modificarán los datos en el sistema y se registrará esta eventualidad en fichero de log.

Una vez importando un fichero Excel conteniendo una lista de agentes, se emitirá un email para cada agente comunicándole que ha sido añadido al Portal de Incidencias, ratificándole su usuario(identificador).

(Opcional) El sistema podría extenderse para emitir cartas en formatos como Word o PDF comunicándole a la persona su nombre de usuario y su clave de acceso.

(Opcional) Si el fichero viniera con errores, se detectarían y se enviarían los datos a un fichero de LOG para su posterior tratamiento.

(Opcional) El analizador de los datos de entrada debe ser configurable, ya que podrían venir los datos en diferentes formatos y no sólo en Excel. Es opcional permitir más de una entrada, pero es obligatorio que el sistema permita en el futuro una ampliación de manera sencilla.

(Opcional) Se puede extender el sistema para contemplar aspectos de seguridad.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 6 de 37

2.2 Agents

Aquellos agentes que puedan conectarse al sistema dispondrán de un servicio web (REST) que permita consultar y obtener información de los agentes que participan en el sistema. Para ello se creará un servicio web muy simple que toma como entrada un fichero JSON que incluye el identificador de usuario, contraseña, y tipo de agente (login/password/kind) según el formato de invocación siguiente:

```
{"login": usuario, "password": password, "kind": tipo de agente}
```

Y en el caso de que dicha combinación aparezca en la base de datos, se devolverá un fichero en formato JSON con la siguiente información:

```
{ "name": Nombre,  
  "location": Coordenadas (opcional),  
  "email": Email  
  "id": identificador,  
  "kind": tipo de usuario,  
  "kindCode": código numérico del tipo de usuario,  
}
```

El campo "kindCode" se obtiene a partir del fichero maestro en formato CSV descrito anteriormente.

(Opcional) Sistema podrá disponer de un subsistema de acceso a través de Web para actualizar la clave de cada usuario.

(Opcional) Se puede extender la funcionalidad anterior para el resto de campos de los usuarios.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 7 de 37

3 Metodología usada

Attribute-Driven Design) (Bass, Clements, & Kazman, 2003) y la norma del SEI/ANSI/IEEE 1471, 2000).

La documentación sigue el esquema propuesto en la guía de aprendizaje de la asignatura y también se han tomado algunas secciones siguiendo las plantillas propuestas en arc42 (<http://arc42.org/>). Las plantillas actuales están en inglés y alemán, pero había una versión anterior con plantillas en español.

Existe un proyecto que usa esas plantillas para documentar una arquitectura de software sencilla sobre una aplicación de gestión de rutas de bicicleta. La documentación está disponible en la Web. Se puede ver aquí:

<http://biking.michael-simons.eu/docs/index.html>

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 8 de 37

4 Identificación de Stakeholders

En este caso los *stakeholders* (personas interesadas) son:

1. Alumnos del grupo E3B1 (Carga de datos).
2. Alumnos del grupo E3B2 (Consulta y gestión de agentes).
3. Administrador del Sistema.
4. Agentes.
5. Profesores de la asignatura.

Así pues, la lista de *stakeholders* queda:

Código	Stakeholder	Intereses (Módulos)
ST-01	Alumnos del grupo E3B1	Carga de ficheros
ST-02	Alumnos del grupo E3B2	Consulta y gestión de agentes
ST-03	Administrador del Sistema	Carga de ficheros
ST-04	Agentes	Consulta y gestión de agentes
ST-05	Profesores de la asignatura	Ambos

Tabla 1. Lista de Stakeholders e intereses

Posteriormente se pasa a describir en más detalle cada uno.

4.1 Alumnos del grupo E3B1

Se trata de los equipos de desarrollo.

Entre sus objetivos están:

- Utilizar tecnologías y metodologías conocidas, minimizando los riesgos relacionados con el aprendizaje de las nuevas.
- Aprender técnicas de desarrollo de software de forma colaborativa y profesional.
- Utilización de tecnologías similares a las del grupo con quien deberán integrarse posteriormente para evitar incompatibilidades.

4.2 Alumnos del grupo E3B2

Equipo de desarrollo del módulo de consulta y gestión de agentes (Agents). Entre sus objetivos están:

- Utilizar tecnologías y metodologías conocidas, minimizando los riesgos relacionados con el aprendizaje de las nuevas.
- Aprender técnicas de desarrollo de software de forma colaborativa y profesional.
- Utilización de tecnologías similares a las del grupo con quien deberán integrarse posteriormente para evitar incompatibilidades.

4.3 Administrador del Sistema

Es la persona que carga los ficheros de datos.

Entre sus objetivos están:

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 9 de 37

- Tecnologías sencillas de los ficheros de entrada.
- Ficheros que puedan leerse por los humanos.
- Ser capaz de automatizar el proceso de carga de listas de agentes.
- Ser capaz de depurar el proceso de carga en caso de errores.

4.4 Agents

Son los usuarios que pueden enviar incidencias al sistema (personas físicas, entidades, sensores, etc).

Entre sus objetivos están:

- Sencillez de acceso a los datos.
- Ser capaz de acceder desde su localización de una forma segura.
- Ser capaz de consultar información del sistema.
- Ser capaz de cambiar su información en el sistema, por ejemplo, la clave (**Opcional**).

4.5 Profesores de la asignatura

Son los responsables de los resultados de la práctica.

Entre sus objetivos están:

- Proponer tecnologías que ayuden a los estudiantes a adquirir habilidades relacionadas con la arquitectura del software mediante el desarrollo de un proyecto práctico.
- Introducir a los estudiantes en el desarrollo de software de forma colaborativa y profesional, mediante desarrollo basado en pruebas (TDD, *test-driven design*).
- Proponer un trabajo de desarrollo a partir de una documentación que pueda realizarse en el tiempo asignado por los estudiantes de la asignatura.
- Mostrar a los estudiantes un ejemplo de documentación de arquitectura.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 10 de 37

5 Atributos de calidad

Para el sistema se han identificado los siguientes atributos de calidad:

- **Disponibilidad**
 - Disponibilidad consultar datos 24x7. datos 24x7.
- **Modificabilidad**
 - Facilidad de cambio de diversas partes de la aplicación: Cambiar el *parser* de entrada de listas de agentes para ficheros con nuevos campos o incluso otro tipo de ficheros.
 - Facilidad de cambio de diversas partes de la aplicación: Añadir nueva información al fichero de *log*
 - Facilidad para modificar partes de la aplicación: otros formatos de salida para las cartas personales
 - Facilidad de cambio de diversas partes de la aplicación: Permitir el cambio de *email*, nombre de usuario...
 - Facilidad para modificar partes de la aplicación: otros formatos, mediante negociación de contenido
- **Rendimiento**
 - El rendimiento del proceso de carga de los ficheros es razonable.
 - La consulta de información de un usuario por el servicio web debe ser rápida.
- **Seguridad**
 - Garantizar la confidencialidad de los datos.
- **Testabilidad**
 - Debe ser posible chequear automáticamente que los datos del fichero se cargan adecuadamente.
 - Debe ser posible chequear que el servicio Web se comporta de forma adecuada.
- **Usabilidad**
 - El sistema de carga de datos debe poder ser usado por usuarios administradores de sistema familiarizados con herramientas tipo Unix.
- **Interoperabilidad**
 - El presente sistema será usado por el Sistema de Análisis de Incidencias, el cual delegará en el sistema actual la gestión de usuarios. El subsistema *Agents* deberá ser utilizado por un proceso automático para consultar el estado de los agentes que quieran enviar incidencias.
- **Simplicidad**
 - Los dos sub-sistemas deberán ser simples y fáciles de desarrollar
- **Desplegabilidad**
 - El sistema debe ser fácilmente desplegable, especialmente en un servidor en la nube.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 11 de 37

5.1 Lista de atributos de calidad

Código	Descripción	Tipo de Atributo	Módulo afectado
AT001	Disponibilidad del sistema para consultar datos 24x7	Disponibilidad	Agents
AT002	Facilidad de cambio de diversas partes de la aplicación: Cambiar el parser de entrada de listas de agentes.	Modificabilidad	Loader
AT003	Facilidad de cambio de diversas partes de la aplicación: Añadir nueva información al fichero de <i>log</i> .	Modificabilidad	Loader
AT004	Facilidad para modificar partes de la aplicación: Añadir otros formatos de salida o de entrada.	Modificabilidad	Loader y Agents
AT005	Facilidad de cambio de diversas partes de la aplicación: Permitir el cambio de contraseña, permitir el cambio de email, permitir el cambio de dirección postal. Permitir cambiar los campos JSON de entrada/salida	Modificabilidad	Agents
AT006	Facilidad para cambiar partes de la aplicación: procesar y devolver información en otros formatos mediante negociación de contenido.	Modificabilidad	Agents
AT007	El rendimiento del proceso de carga de datos de los ficheros es razonable (no demasiado lento, pero tampoco crítico).	Rendimiento	Loader
AT008	El sistema debe garantizar la confidencialidad de los datos de los usuarios	Seguridad	Loader y Agents
AT009	Debe ser posible chequear que el servicio web se comporta adecuadamente	Testabilidad	Agents
AT010	Debe ser posible chequear el comportamiento del sistema de carga de datos .	Testabilidad	Loader
AT011	El sistema debe poder ser usado por administradores de sistemas familiarizados con herramientas tipo Unix	Usabilidad	Loader y Agents
AT012	El servicio Web debe poder ser utilizado por procesos automáticos que consulten el estado de un usuario	Interoperabilidad	Agents
AT013	El sistema debe ser sencillo y fácil de implementar	Simplicidad	Loader y Agents
AT014	El sistema debe ser fácilmente desplegable	Desplegabilidad	Loader y Agents

Tabla 2. Lista de atributos de calidad y tipos

5.2 Atributos de calidad^[OBJ]

Los diferentes atributos de calidad son de interés para alguno de los *stakeholders*. La siguiente tabla muestra la lista de intereses para el proyecto actual:

Atributos vs Interesados	ST-01	ST-02	ST-03	ST-04	ST-05
AT001	X	X		X	X
AT002	X		X		X
AT003	X		X		X
AT004	X	X	X		X
AT005		X			X
AT006	X	X		X	X
AT007	X		X		X
AT008	X	X	X		X
AT009		X	X		X
AT010	X			X	X
AT011		X	X		X
AT012		X		X	X
AT013	X	X		X	X
AT014	X	X	X		X

Tabla 3. Lista de intereses de los *stakeholders*

6 Restricciones

Para realizar esta aplicación existen las siguientes restricciones

6.1 Restricciones técnicas

Código	Restricción	Motivación
TC001	El lenguaje de programación será Java	Se asume que el equipo de desarrollo tiene conocimientos de Java
TC002	Se utilizará una base de datos relacional para almacenar los datos	Se asume que el equipo de desarrollo tiene conocimientos de bases de datos relacionales y existen múltiples librerías para trabajar con bases de datos relacionales desde Java
TC003	El servicio Web estará basado en estilo REST con formato de entrada JSON	El estilo REST es fácil de implementar y consumir.
TC004	Los datos de entrada vienen en formato Excel	Excel es un formato de datos bastante popular y existen varias librerías Java para procesar ficheros Excel
TC005	El formato de salida de los emails personalizadas será texto plano	Con el fin de facilitar la implementación se propone generar cartas personalizadas mediante texto plano. El equipo de desarrollo puede opcionalmente implementar otros formatos
TC007	Pruebas automáticas y desarrollo basado en pruebas	Las pruebas deberán ser ejecutables automáticamente. Se propone un desarrollo basado en pruebas así como la utilización de técnicas de integración continua.
TC008	El servicio Web se implementará mediante el <i>framework</i> Spring Boot	El framework Spring Boot se basa en Spring, que es un <i>framework</i> Java muy popular en la industria. Existen muchos ejemplos y material de ayuda para facilitar el aprendizaje por parte de los estudiantes.

Tabla 4. Restricciones técnicas

6.2 Restricciones organizativas

Código	Restricción	Motivación
OC001	Cada sub-sistema será implementado por un equipo pequeño de estudiantes.	El tamaño de los equipos será de unos 3 ó 4 estudiantes con el fin de que los estudiantes puedan aprender a desarrollar software de forma colaborativa mediante un proyecto simple.
OC002	La estructura de la base de datos será la misma para los 2 sub-sistemas	El pegamento entre los 2 sub-sistemas es la base de datos, cuya estructura debe ser acordada por los 2 equipos.
OC003	El código fuente será gestionado mediante el sistema control de versiones Git en un repositorio público en github	Los sistemas de control de versiones son utilizados por la mayoría de las empresas de desarrollo de software. Github ofrece un software de gestión de proyectos muy potente

Tabla 5. Restricciones organizativas

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 14 de 37

7 Ámbito del sistema y contexto

Para describir la solución se utilizarán diagramas contextuales y texto.

La aplicación está partida en dos procesos:

- Loader: Se encarga de la carga de los ficheros. Utiliza el estilo Batch
- Agents: Se encarga de la consulta y gestión de agentes. Utiliza el estilo micro-servicios.

Ambos se integran usando el arquitectónico de datos compartidos.

En el diagrama de contexto de la Figura 1, se muestran las principales interfaces de cada sistema. El subsistema DataBase es común a ambos grupos, por tanto, hay que acordar la tecnología, el modelo de base de datos y el modo de acceso.

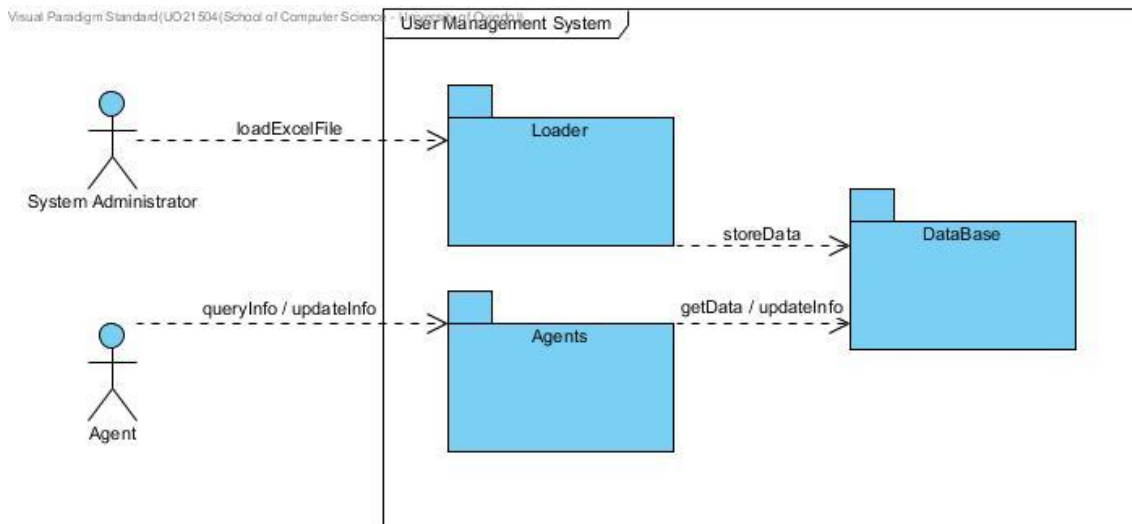


Figura 1. Contexto de negocio del sistema

A continuación, se incluye un diagrama BPMN que define el proceso completo de ambos subsistemas.

A destacar:

- Hay una base de datos común.
- Los datos intercambiados usan formato JSON.
- Los procesos de Loader y de Agents son asíncronos.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 15 de 37

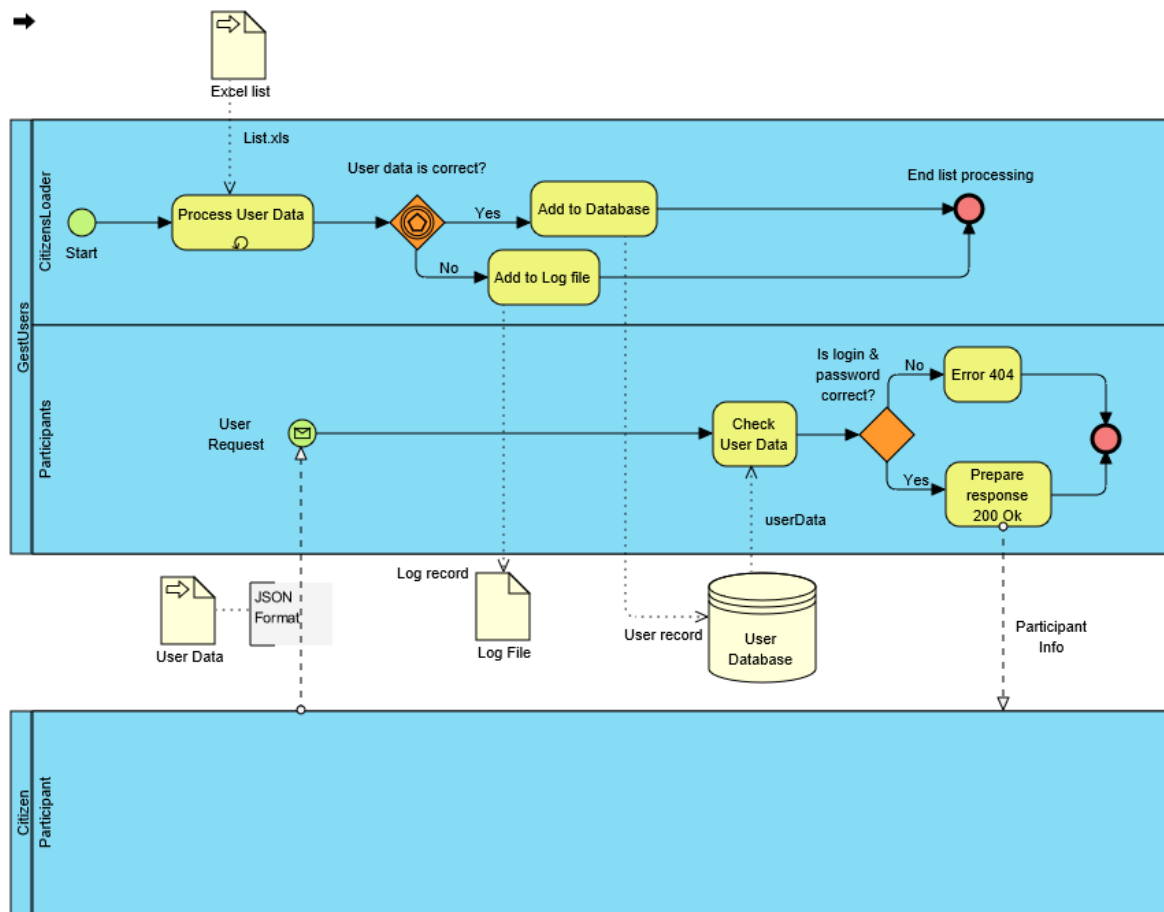


Figura 2. Diagrama BPMN

8 Escenarios de calidad

Con toda la información anterior se procederá a definir los escenarios de calidad que influyen en esta arquitectura.

En las próximas páginas se muestra una tabla con la lista de escenarios identificados.

Escenario Nº	Fuente de estímulo	Estímulo	Entorno	Artefacto	Respuesta	Medición de la respuesta	Atributo de calidad afectado
1	Sistema de participación ciudadana	Realiza una petición de información sobre sus datos	Tiempo de ejecución	Agents	El sistema recibe la respuesta adecuada	La información es recibida en menos de 15 seg. A cualquier hora del día.	AT001
2	Desarrollador	Se introduce un nuevo Parser	Desarrollador	Parser	La modificación es introducida adecuadamente	El sistema es compilado y pasa todas las pruebas	AT002
3	Desarrollador	Se implementan nuevos registros para la generación de informes de error	Desarrollador	ReportWriter, DBUpdate y Parser	La opción es implementada con cambios mínimos que solamente afectan al módulo de generación de informes	Menos de un día de trabajo	AT003
4	Desarrollador	Se añade un nuevo formato de salida	Desarrollador	Agents y DBManagement	Se incluye el nuevo formato con cambios mínimos en el código	Menos de un día de trabajo	AT004
5	Desarrollador	Se introduce la opción de cambio de información de los participantes	Desarrollador	Agents y DBManagement	La información se cambia adecuadamente o se genera el error identificando el problema	Menos de un día de trabajo	AT005
6	Desarrollador	Se añade un nuevo formato a los web-services	Desarrollador	Agents	Se implementa el nuevo formato	Menos de dos días de trabajo	AT006

Escenario Nº	Fuente de estímulo	Estímulo	Entorno	Artefacto	Respuesta	Medición de la respuesta	Atributo de calidad afectado
7	Administrador del Sistema	Cargar una hoja excel en el sistema (DB)	Tiempo de ejecución	Parser, DBUpdate and ReportWriter	Se carga una Excel sin errores en un tiempo razonable	< 1 segundo por cada 10 ciudadanos	AT007
8	Desarrollador	Cargar un sistema en el sistema (DB)	Desarrollo/Tiempo de ejecución	Parser, DBUpdate and ReportWriter (Optional)	La carga debe hacerse de manera segura	No es posible acceder a los datos personales de los usuarios salvo el administrador del sistema, que tampoco puede acceder a las contraseñas.	AT008
9	Participantes	Accede a la aplicación	Tiempo de ejecución	Agents	Cada ciudadano puede acceder a sus datos, pero no a los datos de otros ciudadanos	El acceso a los datos se permite solamente cuando la información de email/contraseña son correctas.	AT009
10	Administrador del Sistema	Carga un fichero Excel en la base de datos	Tiempo de ejecución	Parser, DBUpdate y ReportWriter	El proceso de carga se realiza de una forma fiable y es posible chequear que los datos han sido cargados adecuadamente.	No hay errores en la base de datos ni registros duplicados. Ningún ciudadano tiene menos información que la requerida.	AT010

Escenario Nº	Fuente de estímulo	Estímulo	Entorno	Artefacto	Respuesta	Medición de la respuesta	Atributo de calidad afectado
11	Administrador del Sistema	Carga un fichero Excel en la base de datos	Tiempo de ejecución	Parser, DBUpdate y ReportWriter	El proceso de carga se comporta de una forma habitual y las opciones son fáciles de comprender	El sistema muestra ayuda si el usuario la solicita. Los mensajes de error y otra información son comprensibles por personal técnico	AT011
12	Sistema de Participación Ciudadana	Accede al servicio Web	Tiempo de ejecución	Agents	El sistema de Participación Ciudadana solicita información sobre un usuario pasando una combinación de email y contraseña	Se envía respuesta 200 OK si la combinación aparece en el sistema o error en caso contrario	AT012
13	Desarrollador	Implementa el sistema	Desarrollo	Agents Loader	Los desarrolladores pueden implementar el sistema	El sistema puede implementarse en 2 semanas	AT013
14	Administrador del Sistema	Despliega el sistema	Despliegue	Loader, Agents	El sistema es desplegado en un entorno de producción	El sistema puede desplegarse en menos de una hora	AT014

Tabla 6. Lista de escenarios de calidad

9 Vistas

En los próximos párrafos se describirán algunas de las vistas identificadas y se documentarán de acuerdo a las instrucciones definidas en la guía de aprendizaje.

Vista	Stakeholders	Atributos de calidad	Escenarios
Context	ST-01, ST-02, ST-03, ST-04, ST-05	AT011, AT013	11, 13
Loader	ST-01, ST-02, ST-04, ST-05	AT002, AT003, AT004, AT007, AT008 y AT010, AT011, AT013	2, 3, 4, 7, 8, 10, 11, 13
Agents	ST-01, ST-03, ST-04, ST-05	AT001, AT005, AT006, AT008, AT009, AT012, AT013	1, 5, 6, 8, 9, 12, 13

En los catálogos y en las vistas se han definido tanto los elementos obligatorios como opcionales (voluntarios). Es responsabilidad de cada equipo eliminar los que no vaya a implementar.

9.1 Contexto

La vista de sistema describe los dos subsistemas en interacción, así como sus interfaces.

9.1.1 Presentación principal

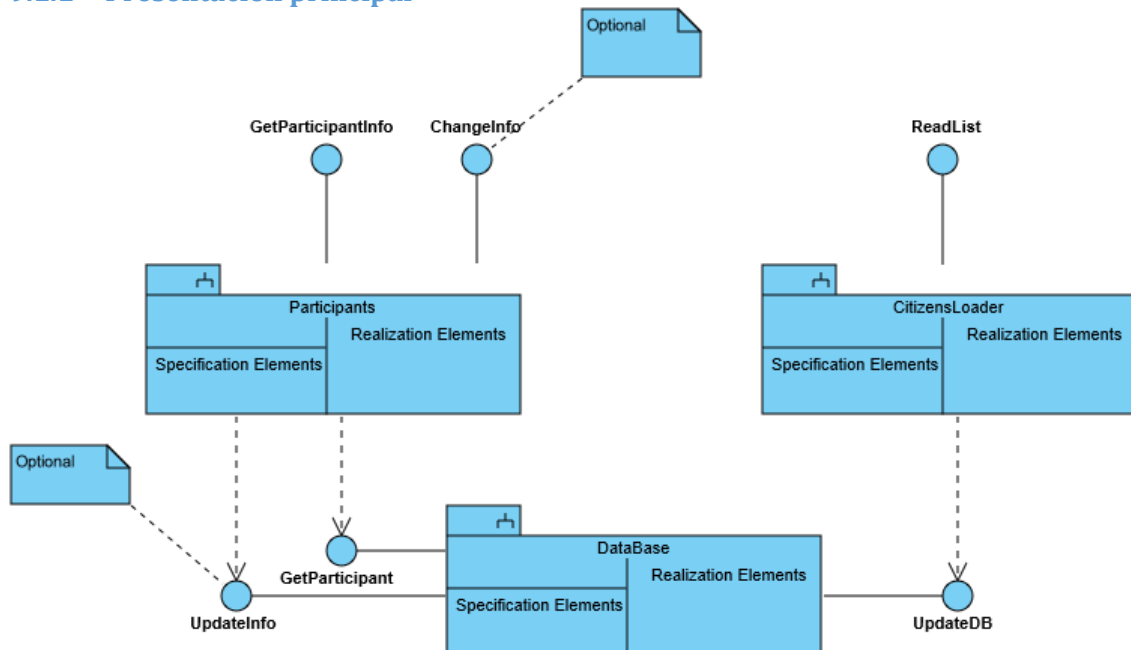


Figura 3. Context view

9.1.2 Catálogo de elementos

9.1.2.1 Elementos

Elemento	Propiedades
Loader	Se encarga de la introducción de las listas de ciudadanos en el sistema. Lee un fichero con

	los datos de los ciudadanos. Crea las claves. Añade los emails para los usuarios dados de alta.
Agents	Es el módulo usado por los agentes para comprobar que han sido dados de alta y opcionalmente para hacer el cambio de clave u otros datos.
DataBase	Este módulo encapsula los accesos a la base de datos.

9.1.2.2 Relaciones

Los datos de los ciudadanos se introducen en el sistema a través de la interface *ReadList* del módulo *Loader*. Para cada usuario, se crea una clave y se emite un email con todos los datos del usuario.

Posteriormente se envían a la base de datos a través de la interfaz *UpdateDB* del módulo *DataBase*.

El módulo *Agents* permite al usuario entrar en sesión a través del servicio web *GetAgentInfo* para comprobar sus datos. Para ello, *Agents* pide los datos al módulo *DataBase* a través de la interfaz *GetAgent*.

Opcionalmente se puede implementar la interface *ChangeInfo* que permite al usuario cambiar su clave u otros datos, para este fin, el módulo *Agents* solicita al módulo *DataBase* el cambio de clave a través de la clase *UpdateInfo*.

9.1.2.3 Interfaces / Puertos

9.1.2.3.1 Loader

Interface	Tipo	Tecnología	Propiedades
ReadList	Interface	Invocación mediante línea de comandos	Se invocará como un programa en consola

9.1.2.3.2 Agents

Interface	Tipo	Tecnología	Propiedades
GetAgentsInfo	Interface	Servicio Web	Este servicio se invocará a través de una petición HTTP

Autores: Alumnos del grupo de prácticas E3B		© 2018	
Escuela de Ingeniería Informática, Univ. Oviedo		Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)			Hoja 21 de 37

9.1.2.3.3 DataBase

Interface	Tipo	Tecnología	Propiedades
GetAgent	Interface	Invocación a Método	Devuelve los datos de un ciudadano.
UpdateDB	Interface	Invocación a Método	Inserta en la base de datos los datos de un ciudadano, incluida su contraseña
UpdateInfo	Interface	Invocación a Método	Actualiza la clave del usuario en la base de datos.

9.1.2.4 Comportamiento

9.1.2.4.1 Loader

Ver el párrafo 9.2.2.3.4.

Además, puede hacer las siguientes opciones:

- **(Opcional)** el subsistema que cree los emails debería implementarse de forma que permita emails o cartas en el futuro en diferentes formatos (Word, ODT, PDF, RTF, ETC.).
- **(Opcional)** Si el fichero viniera con errores, se detectarían y se enviarían los datos a un fichero de LOG para su posterior tratamiento.
- **(Opcional)** El *parser* de los datos de entrada debería ser configurable para permitir datos en diferentes formatos (Excel, TXT, etc.)

9.1.2.4.2 Agents

Permite a los usuarios poder acceder al sistema para comprobar que han sido dados de alta, usando la información recibida en el email. Los usuarios podrían no acceder directamente mediante un navegador Web, sino a través de un sistema externo que invoca el módulo como un servicio Web.

9.1.2.4.3 DataBase

Este módulo encapsulará las operaciones de acceso a la base de datos así como la tecnología a utilizar.

9.2 Loader

La vista de *Citizen Reader* muestra el primer nivel de descripción de los componentes.

9.2.1 Presentación principal

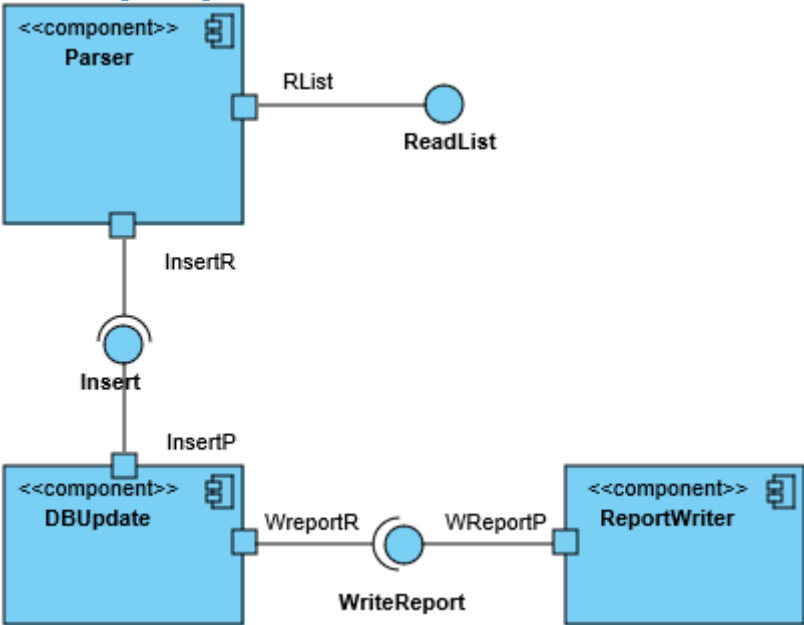


Figura 4. Vista Loader

9.2.2 Catálogo de elementos

9.2.2.1 Elementos

Elemento	Propiedades
Parser	Lee los datos de entrada en Excel y los transforma en un contenedor de objetos que puede ser recorrido para su inserción en la base de datos. También crea el <i>usuario/password</i> del ciudadano y el email usado para la comunicación. Durante el diseño y la implementación hay que partir este componente en los subcomponentes necesarios para separar todos estos servicios y hacerlo de manera que se cumplan los atributos de calidad AT002, AT003, AT004 y AT007.
DBUpdate	Encapsula todas las operaciones de base de datos usando interfaces para permitir el acceso a la base de datos.

Elemento	Propiedades
ReportWriter	Recibe cadenas de información con los datos del usuario que fue imposible de dar de alta y las razones de dicho fallo y escribe un registro en un fichero de texto secuencial, indicando toda la información necesaria para poder revisar visualmente los fallos.

9.2.2.2 Relaciones

El componente *Parser* recibe el fichero de entrada en *Excel* y mediante un *parser* convierte éste en objetos. Añade a éstos objetos el email y el *password*, y lo añade a la base de datos utilizando el componente *DBUpdate*.

Si se producen errores en la carga de datos (DNI duplicados, campo DNI vacío, etc.) o si el componente *de la base de datos* devuelve un error, esta información se escribe en un fichero de LOG mediante la interface *WriteReport* y el componente *ReportWriter*.

(Opcional) Si aparecen otras situaciones de error se pueden documentar usando el mismo componente *ReportWriter*.

9.2.2.3 Interfaces / Puertos

9.2.2.3.1 Parser

Interface	Tipo	Tecnología	Propiedades
ReadList	Interface	Invocación a Métodos	Lee el fichero de <i>Excel</i> con los datos de una lista de ciudadanos.
RList	Port		Crea los subcomponentes del <i>parser</i> necesarios para procesar el fichero de entrada.
Insert	Interface (Requerida)	Invocación a Métodos	Llama a un método del componente <i>DBUpdate</i> para hacer la inserción en la base de datos.
InserR	Port		Verifica los datos y crea el objeto a enviar a <i>DBUpdate</i> .

9.2.2.3.2 DBUpdate

Interface	Tipo	Tecnología	Propiedades
Insert	Interface	Invocación a Métodos	Recibe un objeto con la información para insertar en la base de datos.

Autores: Alumnos del grupo de prácticas E3B		© 2018	
Escuela de Ingeniería Informática, Univ. Oviedo		Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)			Hoja 25 de 37

InsertP	Port		Verifica los datos de entrada y si falta algún atributo obligatorio genera el correspondiente error.
WriteReport	Interface (Requerida)	Invocación a Métodos	Llama a un método del componente <i>ReportWriter</i> para escribir una línea o registro en el fichero de <i>log</i> .
WreportR	Port		Verifica los datos a escribir.

9.2.2.3.3 ReportWriter

Interface	Tipo	Tecnología	Propiedades
WriteReport	Interface	Invocación a Métodos	Recibe los datos para escribir en el fichero de <i>log</i> .
WreportP	Port		Añade a los datos la hora y la fecha.

9.2.2.3.4 Parser

Introduce las listas de agentes en el sistema a partir de ficheros Excel formados por filas de datos, cada una con la siguiente información (excepto la primera fila que contiene las cabeceras):

- Nombre (String)
- Localizacion (clase Localizacion que almacena dos double)
- Email (String con un formato acorde a las convenciones de correo electrónico)
- Identificador (Puede ser el CIF), (String formado por dígitos y letras)
- Tipo(String)

La invocación se hará mediante un programa *batch* ejecutado en línea de comando por el administrador del sistema. Durante la importación las listas de ciudadanos, se creará un usuario por cada ciudadano, cuyo nombre de usuario coincidirá con el correo electrónico y se generará una contraseña aleatoria. La combinación adecuada de email/contraseña permitirá al usuario entrar al sistema, acceder a su información y participar en el portal.

Este componente también creará los emails personales comunicando al usuario que ha sido añadido al Portal de Participación Ciudadana, e informando de su clave de acceso.

9.2.2.3.5 DBUpdate

Actualiza la base de datos. Ver 9.1.2.4.3.

9.2.2.3.6 ReportWriter

Guarda en un fichero de texto la información de los errores producidos en el proceso de conversión. La información básica a guardar es:

- Fecha
- Hora
- Fichero Excel de procedencia
- Descripción del error (con toda la información necesaria)

9.2.3 Diagrama contextual

Ver 9.1.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 26 de 37

9.2.4 Justificación de las decisiones

Las decisiones que han llevado a este diseño son:

Escenario	Atributos de calidad	Justificación
2	AT002	El acceso al <i>parser</i> mediante un patrón <i>Adapter</i> garantiza un cambio rápido de <i>parser</i> sin tocar el código ya realizado en otras partes de la aplicación.
3	AT003	Prever una interfaz y un objeto que pueda estar vacío para el informe de errores (<i>WriteReport</i>) facilita la modificabilidad en caso de añadir nuevos tipos de registros posteriormente.
5	AT005	La utilización de una base de datos relacional ofrecerá un acceso eficiente a la información de los usuarios
6	AT006	Utilizar una base de datos con características de seguridad habilitadas podrá garantizar que los datos están aislados de accesos indebidos. El envío de la contraseña d manera individualizada evita que la información pueda ser leída por otros ciudadanos.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 27 de 37

8	AT008	La utilización de una base de datos relacional con acceso mediante SQL puede permitir a los alumnos verificar que los datos han sido cargados adecuadamente
10	AT010	La utilización de una aplicación <i>batch</i> que pueda ser ejecutada manualmente o configurada para su ejecución automatizada es una práctica común entre los administradores de sistemas
14	AT014	Una aplicación <i>batch</i> independiente puede ser ejecutada directamente sin ninguna necesidad especial para su despliegue

9.3 Agents

9.3.1 Presentación principal

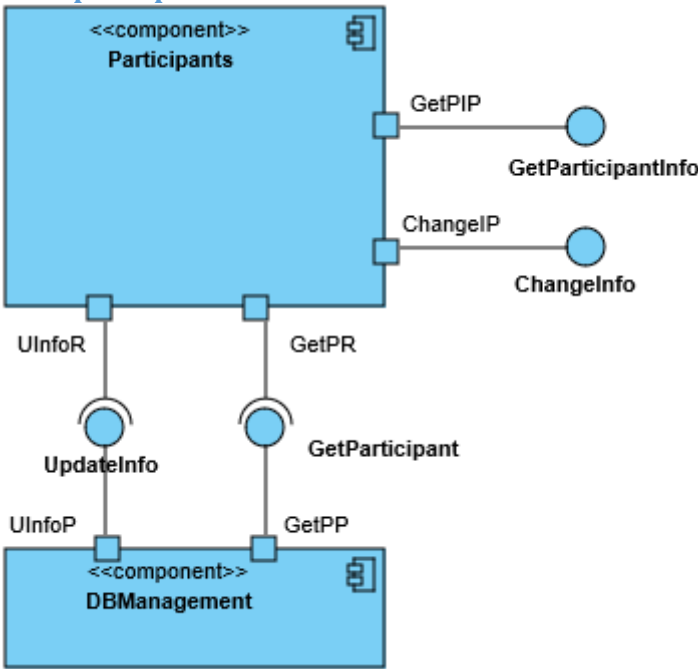


Figura 5. Vista de Agents

9.3.2 Catálogo de elementos

9.3.2.1 Elementos

Elemento	Propiedades
Agents	Se accede a través de dos servicios web: GetAgent, que permite al usuario acceder a sus datos en el sistema y (opcional) ChangeInfo que permite al usuario cambiar su clave de acceso u otros datos.
DBManagement	Se accede a través de dos interfaces: GetAgent, que

devuelve los datos de un ciudadano en la base de datos y (opcional) *UpdateInfo*, para actualizar un cambio de clave en la base de datos.

9.3.2.2 Relaciones

El Sistema de Participación Ciudadana invoca *Agents* utilizando una llamada a un servicio web que es procesada por *GetAgentsInfo*(enviado *email/contraseña*) y éste accede a los datos encapsulados en *DBManagement* mediante la interface *GetAgent*. Si la combinación *email/contraseña* es correcta se devuelve la información del usuario en formato JSON.

(Opcional) El usuario (agente) accede a *Agents* de forma manual al servicio web *ChangeInfo* enviado *usuario/password/newPasswod*) y éste llama a la interface *UpdateInfo* para modificar la clave a través del componente *DBManagement*.

(Opcional) Se pueden crear tantas interfaces como elementos a modificar o usar la anterior con algún tipo de código para definir los datos a modificar.

9.3.2.3 Interfaces / Puertos

9.3.2.3.1 Agents

Interface	Tipo	Tecnología	Propiedades
GetAgentsInfo	Interface	Servicio Web	Permite el acceso a los datos de un participante mediante la combinación <i>email/contraseña</i> .
GetPIP	Port		Hace la validación del usuario antes de pedir los datos.
ChangeInfo	Interface	Servicio Web	Permite el cambio de clave u otra información a un ciudadano mediante una combinación de datos: <i>email/contraseña/nuevaContraseña</i> .
ChangeIP	Port		Hace la validación del usuario antes de solicitar el cambio de clave.
UpdateInfo	Interface (Requerida)	Invocación a Métodos	Solicita el cambio de clave para el usuario.
UInfoR	Port		--
GetAgent	Interface (Requerida)	Invocación a Métodos	Solicita la información para el usuario.
GetPR	Port		--

9.3.2.3.2 DBManagement

Interface	Tipo	Tecnología	Propiedades
UpdateInfo	Interface	Invocación a Métodos	Permite la solicitud de cambio de contraseña u otra información para el usuario.

Autores: Alumnos del grupo de prácticas E3B		© 2018	
Escuela de Ingeniería Informática, Univ. Oviedo		Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)			Hoja 30 de 37

UInfoP	Port		--
GetAgent	Interface	Invocación a Métodos	Permite la solicitud la de información para el usuario.
GetPP	Port		--

9.3.2.4 Comportamiento

9.3.2.4.1 Agents

Ver 9.3.2.2.

Al igual que en el curso pasado, este módulo consiste en un servicio Web REST que permitirá consultar y obtener información de los agentes que participan en el sistema. El servicio Web está pensado para ser utilizado por agentes que puedan conectarse al sistema, por lo que la entrada serán ficheros en formato JSON y las respuestas también serán en formato JSON. Al igual que el año pasado, este sistema podrá disponer de un subsistema de acceso a través de Web para actualizar la clave de cada usuario. Opcionalmente, podrán actualizarse otros campos de los usuarios. El formato de las invocaciones al sistema es el siguiente (obsérvese que se han modificado 2 campos respecto al año pasado):

```
{"login": usuario,
"password": password,
"kind": tipo de agente}
```

En caso de que la combinación login/password/kind aparezca en la base de datos, se devolverá la siguiente información:

```
{ "name": Nombre,
"location": Coordenadas (opcional),
"email": Email "id": identificador,
"kind": tipo de usuario,
"kindCode": código numérico del tipo de usuario, }
```

El campo "kindCode" se obtiene a partir de un fichero maestro en formato CSV que se ha descrito en la sección anterior.

(Opcional) Se puede implementar un interfaz HTML para que el servicio Web pueda también ser utilizado por personas a través de un navegador Web convencional.

(Opcional) El servicio Web puede ser extendido para permitir a los usuarios cambiar su password.

9.3.2.4.2 DBManagement

Encapsula todos los accesos a la base de datos.

9.3.3 Diagrama contextual

Ver 9.1.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 31 de 37

9.3.4 Justificación de las decisiones

Las decisiones que han llevado a este diseño son:

Escenario	Atributos de calidad	Justificación
1	AT001	La utilización de un servicio web REST aprovecha de la tecnología HTTP y facilita el despliegue del sistema en infraestructuras de alta disponibilidad como pueden ser servidores Web, tanto locales como en la nube.
4	AT005	La encapsulación de las características del modelo que afectan a la base durante el desarrollo y la utilización de un framework basado en MVC facilitará el desarrollo de nuevas funcionalidades como las vistas basadas en HTML o el cambio de clave o nuevos servicios de los usuarios.
6	AT006	La utilización del framework Spring Boot facilitará el desarrollo posterior de características comunes de la web como la negociación de contenido, dado que el framework ya contiene herramientas para su implementación.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 32 de 37

Escenario	Atributos de calidad	Justificación
8	AT008	La restricción de acceso mediante <i>email/password</i> se considera suficientemente segura para este proceso. Las claves deberían almacenarse encriptadas.
9	AT009	El desarrollo de un servicio web REST basado en formatos JSON facilitará la creación de pruebas. El framework Spring Boot contiene varias herramientas para pruebas unitarias y de integración.
12	AT012	El uso de un servicio web REST permitirá el acceso automático al sistema a través de software cliente.
13	AT013	El API del servicio web es simple y contiene la funcionalidad mínima necesaria. La utilización del framework Spring Boot facilitará el desarrollo por los estudiantes dado que el framework tiene soluciones para toda la funcionalidad requerida.
14	AT014	La utilización del framework Spring Boot facilita el despliegue. Hay varios ejemplos que muestran

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 33 de 37

Escenario	Atributos de calidad	Justificación
		cómo desplegar aplicaciones basadas en Spring Boot en servidores de producción.

9.4 Vista de Paquetes

9.4.1 Presentación principal

9.4.2 Catálogo de elementos

9.4.2.1 Elementos

9.4.2.2 Relaciones

9.4.2.3 Interfaces / Puertos

9.4.2.4 Comportamiento

9.4.3 Diagrama contextual

Ver ...

9.4.4 Justificación de las decisiones

Las decisiones que han llevado a este diseño son:

Escenario	Atributos de calidad	Justificación
-----------	----------------------------	---------------

9.5 Vista de Despliegue

9.5.1 Presentación principal

9.5.2 Catálogo de elementos

9.5.2.1 Elementos

9.5.2.2 Relaciones

9.5.2.3 Interfaces / Puertos

9.5.2.4 Comportamiento

9.5.3 Diagrama contextual

Ver ...

9.5.4 Justificación de las decisiones

Las decisiones que han llevado a este diseño son:

Escenario	Atributos de calidad	Justificación
-----------	----------------------------	---------------

10 Bibliografía

ANSI/IEEE 1471. (2000). *Recommended Practice for Architectural Description of Software-Intensive Systems*. ANSI/IEEE.

Bass, L., Clements, P., & Kazman, R. (2003). *Software Architecture in Practice, Second Edition*. Boston: Addison Wesley.

Autores: Alumnos del grupo de prácticas E3B		© 2018
Escuela de Ingeniería Informática, Univ. Oviedo	Universidad de Oviedo	Versión 2018.ES.001
GestUsers: Sistema de Gestión de Usuarios Arquitectura Software para GestUsers. Descripción del Primer entregable (2017)		Hoja 37 de 37