



UNIVERSITAT
POLITÀCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Asistente virtual en Telegram para acceder a la
información económica municipal del
Ajuntament de València

Trabajo Fin de Máster

Máster Universitario en Gestión de la Información

Autor: Arnau Campos Albuixech

Tutor: Diego Álvarez Sánchez

2017-2018

RESUMEN

Este trabajo recoge el desarrollo del prototipo de un asistente virtual (*chatbot*) para el programa de mensajería instantánea Telegram que sea capaz de ofrecer la información económica y financiera del Ajuntament de València a la ciudadanía mediante el uso del lenguaje natural.

Este instrumento supone una forma innovadora de acceso a los datos abiertos publicados por las Administraciones Públicas. A su vez, pretende reducir las barreras con las que se encuentra la ciudadanía al acceder a los portales. Para su desarrollo se han usado técnicas de procesamiento del lenguaje natural y tecnologías que permiten la creación de bots para la creación del asistente virtual.

El proyecto se enmarca dentro de la colaboración de la Universitat Politècnica de València y el Ajuntament de València. Así, el resultado alcanzado responde a necesidades actuales tanto de la administración como de la ciudadanía. La cooperación establecida ha permitido realizar el desarrollo en un entorno real siguiendo una perspectiva de diseño centrado en la ciudadanía.

Palabras clave: asistente virtual, chatbot, bot, datos abiertos, Ajuntament de València, gobierno abierto, procesamiento lenguaje natural

RESUM

Aquest treball recull el desenvolupament del prototip d'un assistent virtual (*chatbot*) per al programa de missatgeria instantània Telegram que siga capaç d'oferir la informació econòmica i financera de l'Ajuntament de València a la ciutadania mitjançant l'ús del llenguatge natural.

Aquest instrument suposa una forma innovadora d'accés a les dades obertes publicades per les Administracions Públiques. Al seu torn, pretén reduir les barreres amb què es troba la ciutadania en accedir als portals. Per al seu desenvolupament s'han fet servir tècniques de processament del llenguatge natural i tecnologies que permeten la creació de bots per a la creació de l'assistent virtual.

El projecte s'emmarca dins de la col·laboració de la Universitat Politècnica de València i l'Ajuntament de València. Així, el resultat aconseguit respon a necessitats actuals tant de l'administració com de la ciutadania. La cooperació establerta ha permès realitzar el desenvolupament en un entorn real seguint una perspectiva de disseny centrat en la ciutadania.

Paraules clau: assistent virtual, chatbot, bot, dades obertes, Ajuntament de València, govern obert, transparència, processament de llenguatge natural

ABSTRACT

This work collects the development of one virtual assistance prototype (Chatbot) to support Telegram instant messaging program. Its aim is to provide to the citizens the Ajuntament de València's economic and financial information through a natural language.

This instrument represents an innovative manner to access information and data published by public administrations. Furthermore, it pretends to reduce the existing "barriers" and challenges faced by citizens when they try to have access to different portal or websites. Its development is based on natural language processor and technologies that allow bots programming and create virtual assistants.

The project is in the collaboration framework between the Universitat Politècnica de València and the València City Council (Ajuntament de València). Thus, the result achieved responds to current needs for both, the administration and citizens. The cooperation established has made possible to carry out development in a real environment, following design perspective centered on citizenship.

Key words: virtual assistant, chatbot, bot, open data, Ajuntament de València, open government, transparency, natural language processing

ÍNDICE GENERAL

Índice general	V
Índice de figuras	VII
Índice de tablas.....	VIII
1 Introducción	1
1.1 Motivación	4
1.2 Objetivos	5
1.2.1 Objetivo general	5
1.2.2 Objetivos específicos	5
1.3 Metodología.....	5
1.4 Impacto esperado	6
1.5 Estructura de la memoria	6
2 Estado del arte	8
2.1 Datos abiertos económicos y financieros de los ayuntamientos	8
2.1.1 ¿Qué son los datos abiertos?	8
2.1.2 Marco legal para la apertura y reutilización de datos.....	10
2.1.3 Datos económicos y financieros.....	14
2.1.4 Portal de transparencia y datos abiertos del Ajuntament de València ...	16
2.2 Asistente virtual para acceso a información.....	22
2.2.1 Asistentes virtuales.....	22
2.2.2 Mensajería instantánea	25
2.2.3 Procesamiento del lenguaje natural.....	26
3 Planteamiento de la solución	32
3.1 Análisis del problema.....	32
3.2 Planteamiento de la solución	33
3.3 Metodología.....	34
4 Desarrollo de la solución	36
4.1 Iteración 1	36
4.2 Iteración 2	48
4.3 Iteración 3	53

4.4	Resultados.....	58
5	Análisis de resultados.....	60
6	Conclusiones.....	63
7	Bibliografía	66
	Apéndice A: Prueba de usabilidad del visor presupuestario.....	70
	Apéndice B: Prueba de usabilidad infoecoVLC.....	73
	Apéndice C: Encuesta Google Forms	79
	Apéndice D: Taller.....	87
	Apéndice E: Acto de presentación del proyecto	88

ÍNDICE DE FIGURAS

Figura 2.1. Línea temporal de la aprobación de las normativas sobre datos abiertos y transparencia.....	11
Figura 2.2. Captura del portal de Transparencia y datos abiertos del Ajuntament de València.	19
Figura 2.3. Captura apartado Transparencia.....	20
Figura 2.4. Datasets económicos y financieros del portal de València.	21
Figura 2.5. Visor presupuestario del Ajuntament de València.....	21
Figura 2.6. Relaciones elementos asistente virtual.....	24
Figura 2.7. Fases de procesado del lenguaje natural.	28
Figura 2.8. Conversación simulada de bot con aclaración de conceptos.	29
Figura 3.1. Etapas modelo de prototipado evolutivo.....	34
Figura 4.1. Diagrama de módulos del asistente virtual.....	41
Figura 4.2. Diagrama de procesos de la integración en Telegram.	42
Figura 4.3. Detalle de diagrama de procesos de la integración en Telegram.	43
Figura 4.4. Diagrama de procesos del gestor de diálogos.....	44
Figura 4.5. Diagrama de proceso de cada uno de los ETL.	45
Figura 4.6. Archivos que componen el código fuente.....	46
Figura 4.7. Captura del agente creado en Dialogflow donde se aprecian los Intents creados en la 1ª Iteración.....	47
Figura 4.8. Archivos que componen el código fuente del módulo 3.	47
Figura 4.9. Identificación de la fuente de Información en la iteración 2.	49
Figura 4.10. Primeras cinco líneas del archivo CSV.	50
Figura 4.11. Diseño de la base de datos 2ª iteración.	50
Figura 4.12. Diagrama conceptual ETL 2ª iteración.	51
Figura 4.13. Captura del agente creado en Dialogflow donde se aprecian los Intents creados en la 2ª Iteración.....	52
Figura 4.14. Captura del agente creado en Dialogflow donde se aprecian las Entities creadas en la 2ª Iteración.	52
Figura 4.15. Clasificación de las preguntas obtenidas en el cuestionario de la 3ª iteración.....	54
Figura 4.16. Fragmento de la web donde están disponibles las retribuciones brutas. .	56



Figura 4.17. Diseño de la base de datos 3ª iteración.	56
Figura 4.18. Diagrama conceptual ETL 3ª iteración.	56
Figura 4.19. Captura del agente creado en Dialogflow donde se aprecian los Intents creados en la 3ª Iteración.....	57
Figura 4.20. Capturas de inicio del asistente virtual.	58
Figura 4.21. Vista del asistente en el menú de contactos de Telegram.....	59
Figura A.1. Fotografía del inicio de la sesión donde se realizó la prueba de usabilidad.	70
Figura A.2. Resultados prueba usabilidad visor presupuestario.	72
Figura B.1. Fotografía de la realización de la prueba de usabilidad.....	73
Figura B.2. Formulario seguido en la prueba de usabilidad.	78
Figura C.1. Formulario seguido en la prueba de usabilidad.	80
Figura C.2. Formulario seguido en la prueba de usabilidad página 2.	80
Figura D.1. Fotografía de la realización del taller.....	87
Figura E.1. Fotografía presentación del proyecto.	88
Figura E.2. Fotografía presentación del proyecto.	89

ÍNDICE DE TABLAS

Tabla 2.1. Comparativa características mensajería instantánea.....	26
Tabla 2.2. Comparativa de las características herramientas PLN.	31
Tabla 4.1. Ejemplo de textos que responde el asistente.	59

1 INTRODUCCIÓN

Los avances en el uso de las tecnologías de la información y comunicación (TICs) y el papel que éstas juegan en la administración y la gobernanza han permitido en los últimos años el auge de un modelo de interacción sociopolítica, el Gobierno Abierto (*Open Government*). Este modelo fundamenta sus estrategias sobre 3 pilares: transparencia, participación y rendición de cuentas (*Open Government Partnership*, 2011). Los tres pilares se encuentran interconectados y son necesarios para abrir la acción de gobierno a la ciudadanía.

Nuestro proyecto se centra en el pilar de la transparencia. La transparencia implica el derecho de la ciudadanía a acceder a la información que poseen los gobiernos. Este punto es muy importante para que la ciudadanía pueda ejercer el derecho a la rendición de cuentas y pueda haber colaboración entre la ciudadanía y el gobierno (Nicandro Cruz-Rubio, 2014).

En el germen de cualquier movimiento por la transparencia podemos encontrar la filosofía de datos abiertos (*Open Data*) que persigue, con una ética similar a la del movimiento del software libre, que los datos sean accesibles de forma gratuita con licencias que permitan la reutilización. Además, estos datos deben ser publicados sin procesar y de forma estructurada. Al mismo tiempo que cumplen estos requisitos es recomendable que estén disponibles en línea y que sean legibles por máquinas.

En el Estado español la ley que establece las obligaciones de publicación que afectan a todas las entidades públicas, qué información se debe publicar y cómo, es la “Ley 19/2013, de 9 de diciembre, de transparencia, acceso a la información pública y buen gobierno” que tiene como objetivo ampliar y reforzar la transparencia de la actividad pública, así como regular y garantizar el derecho de acceso a la información relativa a esta actividad. Debido a la organización territorial de España, además de las leyes de ámbito nacional, las autonomías tienen potestad de crear sus propias regulaciones de transparencia. Una de las pioneras en este ámbito fue Galicia, estableciendo la primera regulación sobre transparencia incluso antes que la legislación nacional en 2006. El resto de las autonomías han ido creando sus propias regulaciones progresivamente, siendo el grueso de ellas posteriores a la regulación nacional, centrándose en la ampliación de derechos y deberes en cada territorio. En la Comunitat Valenciana tenemos la “Ley 2/2015, de 2 de abril, Transparencia, Buen Gobierno y Participación Ciudadana”.

Estas leyes contemplan la creación y desarrollo de portales de transparencia. A raíz de esto, muchas administraciones han desarrollado proyectos para crear sus propios portales de transparencia donde publicar los datos que se exigen en la ley. Además, algunas administraciones con la intención de garantizar una transparencia efectiva han

optado por incluir la publicación de datos abiertos en su estrategia de transparencia. Estos datos abiertos serán nuestro ámbito de estudio.

Según el portal datos.gob.es, que organiza y gestiona el Catálogo de Información Pública del sector público de España en estos momentos se encuentran en marcha 293 iniciativas de datos abiertos de las distintas administraciones (Iniciativa Aporta, 2018). Estas iniciativas en su gran mayoría consisten en la elaboración de portales de datos abiertos, los cuales se encargan de publicar datos de forma activa, sin necesidad de que sea previamente solicitada por parte de la ciudadanía y corresponden en gran medida a la única forma de acceso a las publicaciones de las administraciones.

Si analizamos las estadísticas de acceso a los portales, las cifras son muy bajas. En el portal datos abiertos de España (datos.gob.es), que es el nexo de unión de todos los portales de transparencia de las distintas administraciones españolas, tiene como pico de visitas mensuales alrededor de 45.000 (Iniciativa Aporta, 2018). Un número muy pequeño teniendo en cuenta que la población española está alrededor de los 45 millones. Si echamos la vista al portal de València observamos que según la propia información que publica el portal muy pocos *datasets* sobrepasan las 2.500 visitas y prácticamente ninguno sobrepasa las 3.500. Son cifras muy pequeñas para el tamaño de València.

Con estos datos podemos ver que, aunque los datos se ponen a la disposición de la ciudadanía, la ciudadanía no está accediendo. Este hecho, supone una barrera al desarrollo de la transparencia y, a su vez, limita el ejercicio de la rendición de cuentas.

Esta falta de presencia de la ciudadanía en los portales que en principio son para ella, puede deberse a muchos factores, desde factores más sociales, como son la falta de interés por la política, la dificultad para su correcta interpretación o el desconocimiento en cuanto a la existencia de estos; a barreras más técnicas, como el lenguaje técnico empleado (*datasets*, repositorios, APIs, etc.), el exceso de información o la carencia de conocimientos de cómo tratar grandes volúmenes de datos que se publican.

Cada uno de estos problemas se puede abarcar de distintos modos, por ejemplo, se podrían buscar formas de que los datos publicados satisfagan mejor las necesidades de la ciudadanía para solventar la falta de interés. Pero en nuestro ámbito de estudio nos centraremos en las soluciones técnicas que podamos aportar para hacer más accesible la información acercándola a entornos familiares para la ciudadanía.

Una vez decidido esto, se nos plantea la cuestión de conocer en qué entornos se mueve la ciudadanía. Para resolver esta cuestión, lo primero será comprobar desde qué dispositivos accede la ciudadanía a internet, para poder adaptar la solución. Podemos comprobar por el estudio realizado por AIMC (Asociación para la Investigación de Medios de Comunicación) que el 92,1 % de los usuarios reconoce que accede a internet a través de teléfonos móviles superando al 77,5 % de los que acceden a través de

ordenadores portátiles. Pero no solo eso, un 36,9 % lo hace principalmente desde teléfonos móviles superando a ordenadores de sobremesa y ordenadores portátiles. Aunque parezca poco el 36,9 %, un 79,9 % accede a internet casi constantemente o varias veces al día a través de teléfonos móviles (AIMC, 2018).

Con estos datos podemos intuir que una buena solución puede pasar por adecuar el acceso a los datos abiertos a través de smartphones o dispositivos móviles. De nuevo, la cuestión pasa por establecer cómo implementar esta solución. Volvemos a recurrir al estudio de AIMC, cuando preguntan qué servicios utilizan en los teléfonos móviles (tomando como referencia lo realizado el día anterior) el 63,6 % emplean aplicaciones de mensajería instantánea. Siendo esta la tercera función que más se usa, solo superada por navegar por la web y el uso del correo electrónico. Antes de extraer conclusiones vamos a comparar con otro estudio esta última información. El Estudio Anual Redes Sociales de 2017 elaborado por IAB España, que lo consultamos porque además de incluir las redes sociales tradicionales incluye los programas de mensajería instantánea más comunes, nos parece destacable el dato de que el 67 % de usuarios que utilizan redes sociales y programas de mensajería instantánea lo hacen con la intención de chatear o enviar mensajes, siendo la principal tarea que realizan (IAB Spain, 2017).

Con estos datos observamos que una gran parte de la ciudadanía se comunica empleando aplicaciones de mensajería instantánea a través del teléfono móvil. Esto nos hace intuir que tendrán soltura utilizando estas aplicaciones. Aceptando estas premisas, una buena forma de llegar a la gente que no accede a los portales web de datos abiertos siguiendo los métodos tradicionales, sería haciéndolos más accesibles utilizando como interfaz de usuario la propia aplicación de mensajería instantánea. Lo que nos lleva a proponer la realización de un asistente virtual o *chatbot* para hacer que los datos abiertos sean más accesibles mediante un entorno más familiar como es un chat en una aplicación de mensajería instantánea a la cual se puede acceder a través de cualquier *smartphone* o dispositivo móvil.

Para entender qué es un asistente virtual, primero hay que saber qué es un *bot*, ya que un asistente virtual es un tipo de *bot*. Un *bot*, cuyo término proviene de la palabra robot, es un programa informático que realiza tareas automatizadas y repetitivas a través de internet. Estas tareas requerirían mucho tiempo o serían imposibles de realizar por una persona (Techopedia, 2018). En este orden de ideas, un asistente virtual es un *bot* al que se le añade algoritmos de inteligencia artificial para que sea capaz de simular una conversación con una persona. Dentro de la rama del conocimiento de la inteligencia artificial, nos centraremos en el Procesamiento del Lenguaje Natural (PLN) porque nuestra prioridad es que el asistente sea capaz de interactuar frente a entradas textuales.

Existen diferentes plataformas de mensajería y en el marco de este trabajo nos centraremos en el desarrollo para Telegram. Esta decisión está tomada teniendo en



cuenta dos factores principales, el número de usuarios que tiene la plataforma y el soporte que brindan la hora de desarrollar *bots*. Si nos centramos en usuarios por plataforma podemos destacar dos; Whatsapp y Telegram. La primera es utilizada por el 89% de usuarios de teléfonos móviles, la segunda un 15 % (IAB Spain, 2017). Con esta gran diferencia de implantación deberíamos elegir Whatsapp, pero no tiene soporte para crear *bots*, lo que es un gran inconveniente para el desarrollo de este proyecto. A diferencia de Whatsapp, Telegram sí tiene soporte para *bots*, lo que la hace ideal para implementar nuestro asistente.

Debido a los grandes volúmenes de datos e información que liberan las administraciones públicas, en este proyecto nos centraremos en los datos abiertos económicos y financieros del Ajuntament de València. La ventaja de utilizar esta información es que está compuesta por datos estructurados, con los que será más fácil trabajar. Además, este tipo de información suscita un gran interés entre la ciudadanía, por lo que puede favorecer que el producto resultante tenga mejor acogida. El circunscribirnos al municipio de València nos permitirá definir mejor el alcance del proyecto.

1.1 MOTIVACIÓN

Desde que conocí el concepto de Gobierno Abierto me interesó el tema, y con los conocimientos técnicos que he ido adquiriendo en mi formación académica y laboral, me suscitó el interés de cómo utilizar la tecnología para mejorar la calidad democrática de nuestro país.

Este proyecto, en concreto, tuvo comienzo cuando, en torno a la Càtedra Govern Obert se creó una pequeña comunidad interesada en impulsar el desarrollo de soluciones que facilitaran el empoderamiento de la ciudadanía. Dentro de esta comunidad se identificaron algunos retos a los que se les buscaría solución, entre los que se encontraba elaborar asistentes virtuales para acceder a información pública. Este reto sentó las bases del presente trabajo.

Este reto me pareció interesante porque permitía unir conocimientos tecnológicos que me gustaría mejorar, como son el procesamiento del lenguaje natural y la creación y gestión de asistentes virtuales con los conocimientos de gestión de la información aprendidos en el presente máster a fin de crear una solución real.

1.2 OBJETIVOS

1.2.1 Objetivo general

Desarrollar el prototipo funcional de un asistente virtual que sirva para acceder a la información económica ofrecida en la web y el portal de transparencia del Ajuntament de València.

1.2.2 Objetivos específicos

- Probar diferentes herramientas de inteligencia artificial para el reconocimiento de lenguaje natural y evaluar cuál es el más conveniente para el proyecto.
- Incorporar al proceso de desarrollo herramientas de diseño centrado en la ciudadanía para recabar sus opiniones e intereses.
- Implementar una estructura modular del sistema que permita un crecimiento futuro del prototipo.
- Analizar la idoneidad de los asistentes virtuales en Telegram como interfaz de usuario para consulta de información.
- Observar cómo se desenvuelven los ciudadanos al interactuar con el chat al saber que al otro lado hay un robot y no una persona.
- Especificar los requerimientos formales que ayuden en la construcción de una nueva versión del asistente virtual.

1.3 METODOLOGÍA

La metodología seguida en el presente proyecto ha sido diseñada atendiendo a las características específicas del contexto en el que se ha desarrollado el trabajo final de máster. Así, su realización se enmarca en unas prácticas realizadas para la Càtedra Govern Obert. Esta cátedra surge de la colaboración entre la Universitat Politècnica de València y la Concejalía de Transparencia, Gobierno Abierto y Auditoría Ciudadana del Ajuntament de València, con el objetivo de articular iniciativas de transparencia, gobierno abierto y la incorporación de tecnologías digitales para la democracia.

Fruto de lo anterior, la metodología seguida en el presente trabajo final de máster se articula en base a una combinación de trabajo académico con la coordinación con el personal del Ajuntament de València y el personal de la Càtedra Govern Obert. Así, a través del trabajo académico se ha realizado el estudio y análisis del estado en el que se encuentran todos los elementos implicados, la revisión bibliográfica y se ha establecido el marco conceptual.

A su vez, la coordinación con el personal del ayuntamiento ha marcado la metodología de desarrollo seguida. En esta línea, siguiendo un plan de trabajo coordinado se han tomado las decisiones relativas al tipo de ciclo de vida, el alcance de la solución propuesta y el desarrollo del prototipo.

Para terminar, se realiza un análisis crítico de la aplicación desarrollada, indicando los requerimientos formales para que sea posible la continuación del proyecto.

1.4 IMPACTO ESPERADO

Como se comentó en el punto anterior el proyecto surge como colaboración entre la universidad y el Ajuntament de València, por lo tanto, lo que se espera de este trabajo es que el ayuntamiento incorpore la solución como parte de sus políticas transparencia y datos abiertos.

Por otra parte, independientemente de las decisiones del ayuntamiento, se espera que el prototipo funcional resultante del trabajo sea utilizado por la ciudadanía para realizar consultas sobre el área económica y financiera del Ajuntament de València.

En otro orden de ideas, al crear un nuevo canal de comunicación más informal, este nos brinda otro impacto indirecto: el nuevo canal tiene gran potencial para ser escuchado y calibrar de forma más precisa que la actual los intereses de los ciudadanos.

Para terminar, cómo realizaremos el proyecto entero empleando licencias abiertas y el código quedará abierto en la plataforma Github, producimos otro impacto: contribuimos a la generación de conocimiento abierto. Gracias a esto, cualquier ciudadano, organización o investigador podrá empezar sus proyectos a partir de los resultados y conclusiones obtenidas en este.

1.5 ESTRUCTURA DE LA MEMORIA

Después de todo lo planteado, el presente trabajo se organiza siguiendo la siguiente estructura:

1. **Introducción:** Es el capítulo en el que nos encontramos, contiene una breve introducción, la descripción de los objetivos, la metodología del trabajo, el contexto y el impacto esperado.
2. **Estado del arte:** El capítulo se repasa todos los conceptos teóricos necesarios para poder abordar el trabajo y realiza un estudio del estado actual de las tecnologías implicadas en este proyecto.

3. **Planteamiento de la solución:** En el capítulo se analiza el problema a abordar con la solución que proponemos nosotros. Además, se define la metodología que se seguirá para el desarrollo.
4. **Desarrollo de la solución:** Explicación detallada del proceso llevado a cabo para realizar la solución y el resultado del desarrollo.
5. **Análisis de resultados:** Contienen un análisis crítico de todo el proyecto realizado y los requerimientos formales que deberán tenerse en cuenta en futuros desarrollos a partir del proyecto.
6. **Conclusiones:** Repaso a las conclusiones obtenidas de la realización de este proyecto.
7. **Bibliografía y apéndices.**

2 ESTADO DEL ARTE

En este capítulo daremos un repaso a todos los conceptos teóricos necesarios para poder abordar el trabajo propuesto. Por un lado, abordamos el estudio de los datos abiertos económicos y financieros de los ayuntamientos. Por otro, profundizaremos el análisis de los asistentes virtuales para acceder a esta tipo información. A su vez, realizaremos un estudio del estado actual de las tecnologías implicadas en este proyecto.

2.1 DATOS ABIERTOS ECONÓMICOS Y FINANCIEROS DE LOS AYUNTAMIENTOS

En este apartado haremos una revisión teórica de los datos abiertos económicos y financieros de los ayuntamientos. Para comenzar realizaremos un repaso al significado de los datos abiertos. Seguidamente, abordaremos el marco legal que rodea la reutilización de la información y la transparencia, tanto a nivel nacional como supranacional. Luego profundizaremos sobre lo que son, en el marco del presente proyecto, los datos económicos y financieros y, para terminar, revisaremos cómo se publica esta información en los portales de datos abiertos y, en concreto, en el del Ajuntament de València.

2.1.1 ¿Qué son los datos abiertos?

En la Era de la Información los datos desempeñan un papel esencial en el desarrollo económico y social. Son la nueva materia prima sobre la que se construye el conocimiento para las empresas, los gobiernos y la ciudadanía. En la actualidad se publica una gran cantidad de datos sobre muchas temáticas; ciencia, estadística, medio ambiente, finanzas, etc. Las tareas de recopilación y distribución son realizadas por organismos privados, administraciones públicas e incluso por la ciudadanía¹. En nuestro caso, los que nos suscitan más interés son los datos producidos o recopilados por las administraciones públicas (*Open Government Data*).

La mera publicación de los datos no garantiza la posibilidad de uso o reutilización de estos. Para garantizar que además de publicarlos se puedan reutilizar y generar más conocimiento a partir de su uso, surge el movimiento de los datos abiertos. Al igual que otros movimientos de apertura como el software libre o el código abierto, la filosofía de

¹ Para ampliar información sobre los datos generados por la ciudadanía desde el contexto valenciano: <http://www.valencia.es/transparencia/publicaciones/DATOS%20GENERADOS%20POR%20LA%20CIUDADANIA%20DESDE%20EL%20CONTEXTO%20VALENCIANO.pdf>. Último acceso: 04 septiembre 2018.

datos abiertos (*Open Data*, en inglés) busca que los datos estén disponibles sin restricciones de copyright.

Para entender mejor qué son los datos abiertos acudiremos a lo que dicen dos organismos: el *Open Knowledge International* (OKI a partir de ahora), que es una organización global sin ánimo de lucro enfocada en promover la filosofía de los datos abiertos en la sociedad civil; y el *Open Government Partnership* (OGP a partir de ahora), que es una organización internacional que promueve el gobierno abierto entre los países miembros.

En el *Open Data Handbook* de OKI se definen así los datos abiertos:

“Los datos abiertos son datos que pueden ser utilizados, reutilizados y redistribuidos libremente por cualquier persona, y que se encuentran sujetos, cuando más, al requerimiento de atribución y de compartirse de la misma manera en que aparecen.”
(Open Knowledge International, 2012).

La otra definición la podemos encontrar en la Carta Internacional de los Datos Abiertos de OGP:

“Datos abiertos son datos digitales que son puestos a disposición con las características técnicas y jurídicas necesarias para que puedan ser usados, reutilizados y redistribuidos libremente por cualquier persona, en cualquier momento y en cualquier lugar.”
(Open Government Partnership, 2015).

La definición proporcionada por la Carta Internacional de los Datos Abiertos también incluye los 6 principios que forjan la base del acceso, la publicación y el uso de los datos. Estos principios indican que los datos deben ser:

1. **Abiertos por defecto.** Esto quiere decir que hay que cambiar el modo en el que se entiende la publicación de datos. En vez de que los datos estén cerrados y cada petición de publicación tenga que evaluarse si se debe publicar o no, todos los datos serán publicables salvo los que de forma justificada por razones de seguridad o protección de datos se puedan mantener cerrados.
2. **Oportunos y exhaustivos.** Los datos abiertos tienen que ser relevantes, actualizados, comprensibles y en la medida de lo posible deben abrirse en su forma original y sin modificaciones para que sean valiosos.
3. **Accesibles y utilizables.** Los datos deben ser gratuitos bajo una licencia libre, tienen que ser legibles por máquina y además deben ser fáciles de encontrar. Hay que cuidar los formatos en los que se publica la información y eliminar las barreras administrativas que impiden a los ciudadanos acceder a los datos.

4. **Comparables e interoperables.** Los datos deben estar en formatos estructurados y estandarizados que garanticen la interoperabilidad, trazabilidad y reutilización efectiva. Además, para que tengan más potencial debe ser posible comparar entre distintos datos a través de localizaciones geográficas y el tiempo.
5. **Para mejorar la gobernanza y la participación ciudadana.** Los datos abiertos deben servir para aumentar la transparencia, lo cual aumentará la capacidad de que los ciudadanos tengan una mejor idea de lo que hace la administración, y por ende reforzará la gobernanza y la confianza en las instituciones públicas.
6. **Para el desarrollo incluyente y la innovación.** Aunque con la liberación de datos abiertos no basta para promover la innovación y el desarrollo, si pueden servir para que los actores que lo necesiten tengan los datos necesarios para ayudar a identificar desafíos sociales y económicos, así como monitorear y llevar a cabo programas de desarrollo sostenible.

Después de todo lo visto asumimos la definición de los datos abiertos incluida en la Carta Internacional de los Datos Abiertos y es a la que nos ceñiremos en el resto del documento.

2.1.2 Marco legal para la apertura y reutilización de datos

La legislación a favor de la apertura y reutilización de datos públicos, así como la legislación de transparencia, ha ido mejorando y ampliando derechos según evoluciona políticamente la democracia española, influida en ocasiones por legislaciones europeas que pretendían brindar estos derechos y armonizarlos en todos los países miembros.

En el ámbito nacional, el antecedente legislativo que marcará toda la legislación futura llega con la implantación de la democracia en España y la Constitución de 1978. Empiezan los cambios que permitieron allanar el camino tanto para el desarrollo del derecho al acceso a la información, como el de participación, piedras angulares para la apertura de datos.

El **acceso a la información** a través del artículo 105.b) que contempla el acceso a la información pública por parte de la ciudadanía, salvo que afecte a la seguridad y defensa del Estado, la investigación de los delitos y la intimidad de las personas. Y el **derecho de participación** otorgado por el artículo 23: “Los ciudadanos tienen el derecho a participar en los asuntos públicos, directamente o por medio de representantes, libremente elegidos en elecciones periódicas por sufragio universal.” (Constitución Española, 1978).

Teniendo claro estos antecedentes, haremos un repaso a la legislación más relevante de aplicación en España. El repaso se hará, como se ve en la Figura 2.1, en orden

cronológico y agrupadas según el tipo de legislación: europea, nacional, autonómica y local.



Figura 2.1. Línea temporal de la aprobación de las normativas sobre datos abiertos y transparencia.
Fuente: Elaboración propia.

Leyes europeas

Directiva 2003/98/CE

La Directiva 2003/98/CE del Parlamento Europeo y del Consejo, de 17 de noviembre de 2003, relativa a la reutilización de la información del sector público establece un conjunto mínimo de normas para la regulación de la reutilización, e incorpora instrumentos que facilitan la reutilización de los documentos de los organismos del sector público de los estados miembros. Entendiendo como documentos “cualquier contenido sea cual sea el soporte (escrito en papel o almacenado en forma electrónica o como grabación sonora, visual o audiovisual); o cualquier parte de tal contenido” (Directiva 2003/98/CE, 2003).

Esta directiva no obliga a la publicación ni a autorizar la reutilización de los documentos publicados, pero homogeneiza las condiciones en las que se hará. Algunas de las cuestiones que homogeniza son el tratamiento de solicitudes de información, los formatos aplicables, la tarificación en el caso de haberla, las licencias permitidas y la prohibición de los acuerdos exclusivos con terceros.

Directiva 2013/37/UE

La Directiva 2013/37/UE del Parlamento Europeo y del Consejo, de 26 de junio de 2013, por la que se modifica la Directiva 2003/98/CE relativa a la reutilización de la información del sector público. Esta modificación se debe al cambio del contexto desde la anterior directiva, después de diez años el volumen de datos se ha incrementado exponencialmente y la tecnología para el análisis, explotación y tratamiento de datos está en constante evolución. Esto hace que la anterior norma quede desfasada respecto a estos rápidos cambios, con la intención de paliar esto se incluyen una serie de cambios, entre los cuales los más destacados son (Directiva 2013/37/UE, 2013):

- Introduce la obligación inequívoca a los estados de autorizar la reutilización de todos los documentos, salvo si el acceso está restringido o excluido.

- Amplía el ámbito de aplicación a bibliotecas, museos y archivos.
- Establece que siempre que sea posible los documentos publicados deben ser con formatos abiertos y legibles por máquina junto con sus metadatos, con los niveles más elevados de precisión y granularidad, en un formato que garantice su interoperabilidad.

Leyes nacionales

Ley 37/2007

La Ley 37/2007, de 16 de noviembre, sobre reutilización de la información del sector público tiene como objeto adaptar la Directiva 2003/98/CE para regular el régimen jurídico aplicable a la reutilización de los documentos elaborados o custodiados por las Administraciones y organismos del sector público. Entendiendo documentos en el sentido más amplio, siendo toda información sea cual sea su soporte y su forma de expresión utilizada.

Además de lo establecido en la Directiva 2003/98/CE podemos destacar los siguientes aspectos que introduce esta ley (Ley 37/2007, 2007):

- Las administraciones promoverán que tanto la publicación de documentos como las solicitudes de reutilización se realicen por medios electrónicos, y estos medios electrónicos serán accesibles a las personas con discapacidad.
- Prevé que se establecerán licencias tipo para determinar las condiciones de reutilización.
- La ley establece un régimen sancionador ante el mal uso de la información publicada.

Real Decreto 1495/2011

Después de los primeros años de vigencia de la Ley 37/2007 se detectaron algunas necesidades de desarrollo normativo en el sector público estatal, por lo que se suplieron las necesidades con el Real Decreto 1495/2011, de 24 de octubre. Este Real Decreto sólo era de aplicación en la Administración General del Estado, pero sirvió como modelo para concretar algunas estrategias autonómicas y locales.

El Real Decreto 1495/2011 toma como base la Ley 37/2007 y entre otras se establecen las obligaciones de la administración a la hora de promover la reutilización. Algunas de estas obligaciones son: que generaliza la autorización de reutilización siempre que no haya restricciones; y la obligatoriedad de las administraciones informar de los documentos reutilizables, preferiblemente a través de sedes electrónicas.

También se añade una disposición adicional que insta a crear una normativa técnica relativa a la reutilización de recursos de información que dará lugar a la creación de la

Norma Técnica de Interoperabilidad de reutilización de recursos de la información. (Real Decreto 1495/2011, 2011)

Norma Técnica de Interoperabilidad de reutilización de recursos de la información (NTI)

Las normas técnicas tienen como objeto dar respuesta a las necesidades técnicas derivadas de una Ley. En este caso la Norma técnica de interoperabilidad de reutilización de recursos de la información (NTI a partir de ahora) de 19 de febrero de 2013, tiene como objeto “establecer el conjunto de pautas básicas para la reutilización de documentos y recursos de información elaborados o custodiados por el sector público a los que se refiere el artículo 3 de la Ley 37/2007, de 16 de noviembre, sobre reutilización de la información del sector público por cualquier agente interesado.” (Norma Técnica de Interoperabilidad, 2013)

Ley 19/2013

La Ley 19/2013, de 9 de diciembre, de transparencia, acceso a la información pública y buen gobierno, no regula la reutilización como las anteriores leyes. Esta Ley tiene como objetivo “ampliar y reforzar la transparencia de la actividad pública, regular y garantizar el derecho de acceso a la información relativa a aquella actividad y establecer las obligaciones de buen gobierno que deben cumplir los responsables públicos, así como las consecuencias derivadas de su incumplimiento.” (Ley 19/2013, 2013).

Este objetivo lo cumple a través de los siguientes tres ángulos:

- Transparencia de la actividad pública: con la intención de reforzar la transparencia amplía las obligaciones de publicidad activa.
- Derecho del acceso a la información pública: regula el derecho reconociendo y garantizando el acceso a la información.
- Buen gobierno: enumera una lista de principios que deben cumplir los responsables públicos, y establece sanciones para garantizar la correcta aplicación.

A raíz de esta Ley las Comunidades Autónomas y Entidades Locales dispusieron de un plazo de dos años desde la aprobación para que adapten sus legislaciones a las obligaciones enunciadas por esta.

Ley 18/2015

La Ley 18/2015, de 9 de julio, por la que se modifica la Ley 37/2007, de 16 de noviembre, sobre reutilización de la información del sector público incorpora a la legislación española las modificaciones introducidas por la Directiva 2013/37/EU del Parlamento



Europeo y del Consejo, de 26 de junio de 2013 a la anterior Ley de Reutilización de Información Pública (Ley 18/2015, 2015).

Leyes autonómica y local en el contexto valenciano

Ley 2/2015 Autonómica Valenciana

La Ley 2/2015, de 2 de abril, de Transparencia, Buen Gobierno y Participación Ciudadana de la Comunitat Valenciana marca por primera vez en la normativa autonómica amplias obligaciones de transparencia de la actividad pública, derecho acceso a la información y buen gobierno.

Con esta ley se adapta la Ley nacional 19/2013, de 9 de diciembre, de transparencia, acceso a la información pública y buen gobierno a la Comunidad Valenciana. Aparte de adaptar lo que establece la anterior ley, en algunos apartados desarrollan más lo establecido para mejorar la transparencia, el derecho al acceso y el buen gobierno (Ley 2/2015, 2015).

Regulación local de València

En la ciudad de València no existe una ordenanza que regule la transparencia, aunque en la actualidad están realizando las consultas públicas con carácter previo a la elaboración de esta normativa municipal².

En cambio, sí disponen de un reglamento de transparencia y participación ciudadana desde el 28 de septiembre de 2012, con una modificación aprobada el 24 de abril de 2015. El reglamento son normas de carácter interno con la finalidad de organizar el funcionamiento de la corporación, a diferencia de las ordenanzas que contienen disposiciones generales dirigidas a los ciudadanos.

Con este reglamento el ayuntamiento pretende establecer los cauces de comunicación, información y participación de la ciudadanía. Para ello desarrolla los mecanismos de participación a través del derecho de acceso a la información pública, archivos y registros, derecho de participación y derechos de iniciativa ciudadana. También regula toda la información que se publicará de forma activa por parte del ayuntamiento y cómo (Reglamento de transparencia y participación ciudadana , 2015).

2.1.3 Datos económicos y financieros

A nivel nacional la Norma Técnica de Interoperabilidad de Reutilización de Recursos de la Información establece en el apartado III cómo seleccionar la información que va a ser

² Más información de la consulta en:

<https://www.valencia.es/ayuntamiento/ayuntamiento.nsf/vDocumentosTituloAux/067575EB137DE552C125828100410DE0?OpenDocument&bdOrigen=ayuntamiento%2Fayuntamiento.nsf&idapoyo=967B0E31C3940650C12576700045950A&lang=2&nivel=4>. Último acceso: 04 septiembre 2018.

reutilizada. Como esta norma la tienen que aplicar organismos de muy diversa índole solo detalla unas directrices generales. La selección de la información recae en el organismo que gestiona los datos, que después de identificar sus recursos establecerá las prioridades será según su criterio subjetivo de mayor relevancia y potencial social y económico. Esta información será en la medida de lo posible no alterada y publicada de en su estado original. También será publicada en el nivel granular más fino posible de forma estructurada para permitir el procesamiento automatizado (Norma Técnica de Interoperabilidad, 2013).

A parte de esto también incluye la taxonomía de sectores primarios que servirá para categorizar los catálogos de recursos. Establece veintidós categorías a través de las cuales se deben distribuir la información publicada en los portales de datos abiertos dentro de las cuales se encuentran economía, hacienda y sector público que son materia de estudio de este proyecto. En economía podemos encontrar información económica y financiera de todo tipo, en hacienda podemos encontrar toda la información relacionada con tributación y en sector público se puede encontrar información muy diversa ya que abarca todo lo relativo a los organismos, pero los que nos resultan de interés son los presupuestos, las contrataciones y las licitaciones, ya que el resto de información incluida en esta categoría no contiene información económica y financiera.

Pero para saber qué datos exactamente podemos encontrar publicados por parte del Ajuntament de València, que es con la información que vamos a trabajar, acudiremos al Reglamento de transparencia y participación ciudadana. Este reglamento regula dentro del capítulo III la publicidad activa, qué información debe publicar cada uno de los organismos compuestos por el sector público local del ayuntamiento de València y del artículo 18 al 23 especifica qué información debe ser publicada para cada uno de los seis grupos en los que agrupa la información. Los grupos son los siguientes (Ajuntament de València, 2017):

1. Información sobre la institución, organización, planificación y personal.
2. Información de altos cargos de la entidad.
3. Información de relevancia jurídica y patrimonial.
4. Información sobre contratación, convenios y subvenciones.
5. Información económica, financiera y presupuestaria.
6. Información sobre servicios y procedimientos.

En nuestro proyecto solo es de interés la información contenida en el punto cinco: Información económica, financiera y presupuestaria, en la que detalla que se publicará la información relativa a:

- a) Los presupuestos anuales, con descripción de las principales partidas presupuestarias e información actualizada al menos trimestralmente sobre su estado de ejecución.
- b) Las modificaciones presupuestarias realizadas.
- c) Información sobre el cumplimiento de los objetivos de estabilidad presupuestaria y sostenibilidad financiera y regla de gasto.
- d) La liquidación del presupuesto.
- e) Las cuentas anuales que deban rendirse y los informes de auditoría de cuentas y de fiscalización por parte de los órganos de control externo que sobre dichas cuentas se emitan.
- f) Masa salarial del personal laboral del sector público local, en los términos regulados en el artículo 103 bis de la Ley 7/1985, de 2 de abril.
- g) Datos de endeudamiento mensual, informes de morosidad, e informes sobre planes de ajuste.

(Ajuntament de València, 2017)

2.1.4 Portal de transparencia y datos abiertos del Ajuntament de València

Los portales de datos abiertos son el elemento esencial que da soporte tecnológico a las estrategias de apertura de datos. Aquí se mantendrán publicados todos los datos que se publiquen por parte del organismo, además también suelen incluir visualizaciones de los datos para que ciudadanos con menos conocimientos técnicos puedan entender mejor la información.

Las estructuras de estos portales son siempre muy parecidas, y suelen disponer como mínimo de los siguientes componentes (Federación Española de Municipios y Provincias, 2017):

- *Datasets*: Conjunto de datos que son publicados por la administración para que puedan ser reutilizados.
- *Catálogo*: Este catálogo permitirá a los ciudadanos y empresas identificar de forma sencilla toda la información disponible en el portal. Incluye todos los *datasets* publicados.

- Buscador: Que permita la búsqueda por términos (palabras clave).
- Opciones de filtrado: Que permita filtrar por los sectores primarios en los que están clasificados los *datasets*, por formatos disponibles, por frecuencia de actualización o cualquier otra que se considere necesaria.
- Condiciones legales de uso y reutilización.
- Herramientas que permitan la colaboración ciudadana básica, como lo es un medio de contacto entre la administración y el reutilizador que permita tanto la petición de ayuda técnica como la petición de nuevos datos.
- Una API (*Application Programming Interface*) que permita facilitar el acceso y reutilización de los datos.

Además de los componentes básicos es recomendable para que puedan ser más útiles para la ciudadanía algunos componentes más:

- Herramientas de visualización, a parte de los *datasets* en bruto para que cada ciudadano pueda trabajar con ellos y generar conocimiento a partir de ellos, es recomendable que por parte de la administración que se encarga de publicar los datos ofrezca visualizaciones interactivas para que ciudadanos sin conocimientos técnicos de reutilización de datos pueda acceder y entender la información publicada.
- Ejemplos de uso, elaborar un archivo de ejemplos de éxito en la reutilización de la información publicada
- Herramientas que permitan la colaboración ciudadana avanzada: como son crear un archivo de aplicaciones realizadas a partir de la reutilización de los datos por parte de la ciudadanía, permitir que el portal contenga *datasets* generados por la ciudadanía.

Después de los componentes de los portales de datos abiertos vamos a centrarnos en el componente principal, los datos. Los datos pueden publicarse en múltiples formatos, pero dependiendo de cuál sea empleado marcará en gran medida su posible reutilización.

Los formatos de los datos pueden ser reutilizables o no reutilizables y abiertos o propietarios. Los formatos reutilizables son aquellos que permiten que una máquina pueda leer y procesar automáticamente, mientras que los no reutilizables no. Los datos no reutilizables no se consideran datos abiertos. Por otro lado, los formatos abiertos son aquellos que están formados por una estructura conocida de forma pública y su estructura / especificación no contiene restricciones legales y económicas de uso.

Para que sean datos abiertos tienen que tener formatos reutilizables y preferiblemente tienen que ser abiertos. Los formatos abiertos que más se usan a la hora de publicar los datos abiertos son:

- **CSV** (*Comma-Separated Values*)³: Documento que representa los datos en tablas, donde las columnas se separan por comas y las filas por saltos de línea.
- **XML** (*eXtensible Markup Language*)⁴: Desarrollado por W3C se trata de un lenguaje general de etiquetado, que define un conjunto de reglas para estructurar la información.
- **JSON** (*JavaScript Object Notation*)⁵: Es un subconjunto de la notación literal de objetos de JavaScript, aunque hoy en día se considera un formato independiente. Es liviano para intercambiar datos, fácil de leer por humanos y fácil de interpretar y generar por máquinas.
- **RDF** (*Resource Description Framework*)⁶: Diseñado por W3C se emplea para añadir información semántica siguiendo los fundamentos de la web semántica.
- **RSS** (*Really Simple Syndication*)⁷: Se trata de una estructura XML que permite la distribución de contenidos y actualizaciones de forma muy simple.
- Algunos formatos específicos para información georreferenciada:
 - geoJSON
 - WMS
 - WFS
 - KML
 - SHP
 - geoRSS

Además de los formatos abiertos también se emplean formatos propietarios, aunque no sea recomendable. El formato propietario más destacable en la publicación de datos abiertos es el **XLS** o **XLSX**, que se trata del formato propietario de Microsoft en el que almacena hojas de cálculo.

También es importante destacar un formato que, aunque no es reutilizable se está utilizando mucho para liberar información por parte de organismos. El formato es el **PDF**

³ Más información sobre CSV: <https://tools.ietf.org/html/rfc4180>. Último acceso: 7 agosto 2018.

⁴ Más información sobre XML: <https://www.w3.org/XML/>. Último acceso: 7 agosto 2018.

⁵ Más información sobre JSON: <https://www.json.org/>. Último acceso: 7 agosto 2018.

⁶ Más información sobre RDF: <https://www.w3.org/RDF/>. Último acceso: 7 agosto 2018.

⁷ Más información sobre RSS: <http://www.rss.nom.es/>. Último acceso: 7 agosto 2018.

(*Portable Document Format*) que es un formato abierto que contiene información, pero esta información no se puede reutilizar de forma directa.

Después de ver como son los portales de datos abiertos y los formatos en los que podemos encontrar los datos, vamos a ver el caso del portal de datos abiertos del ayuntamiento de València. En la Figura 2.2 se muestra la ventana de inicio del portal, que aúna el portal de transparencia y el portal de datos abiertos en el mismo sitio web.



Figura 2.2. Captura del portal de Transparencia y datos abiertos del Ajuntament de València.
Fuente: <http://gobiernoabierto.valencia.es/va/>. Último acceso: 1 agosto 2018.

En la figura se puede observar que destacan cuatro apartados en los que está estructurado el sitio web:

- **Transparencia:** En este apartado se podrá acceder a toda la información relacionada con la actividad municipal, tanto la obligada por la normativa como la que el ayuntamiento quiera para incrementar su transparencia. La mayoría de la información que se puede encontrar en este apartado no está en formatos reutilizables.
- **Buen gobierno:** Incluye todo lo relativo al Código de Buen Gobierno de la Generalitat Valenciana y todas las adhesiones por parte del ayuntamiento a él.
- **Datos abiertos:** Accede a un portal de datos abiertos con las características mencionadas anteriormente.
- **Visor presupuestario:** Representa de forma gráfica la información presupuestaria municipal.

Después de ver cómo se estructura el portal del Ajuntament de València, vamos a identificar la información económica y financiera que podemos encontrar en los distintos apartados.

En el apartado de Transparencia podemos encontrar el listado que se observa en la Figura 2.3. La lista está organizada en información presupuestaria, financiera y las cuentas anuales y los informes de fiscalización, pero la información no está almacenada en este sitio web y cuando queremos acceder a algo nos redirige a el sitio web del Ajuntament de València donde se encuentra alojada la información generalmente en formatos no reutilizables como es el caso del formato PDF.

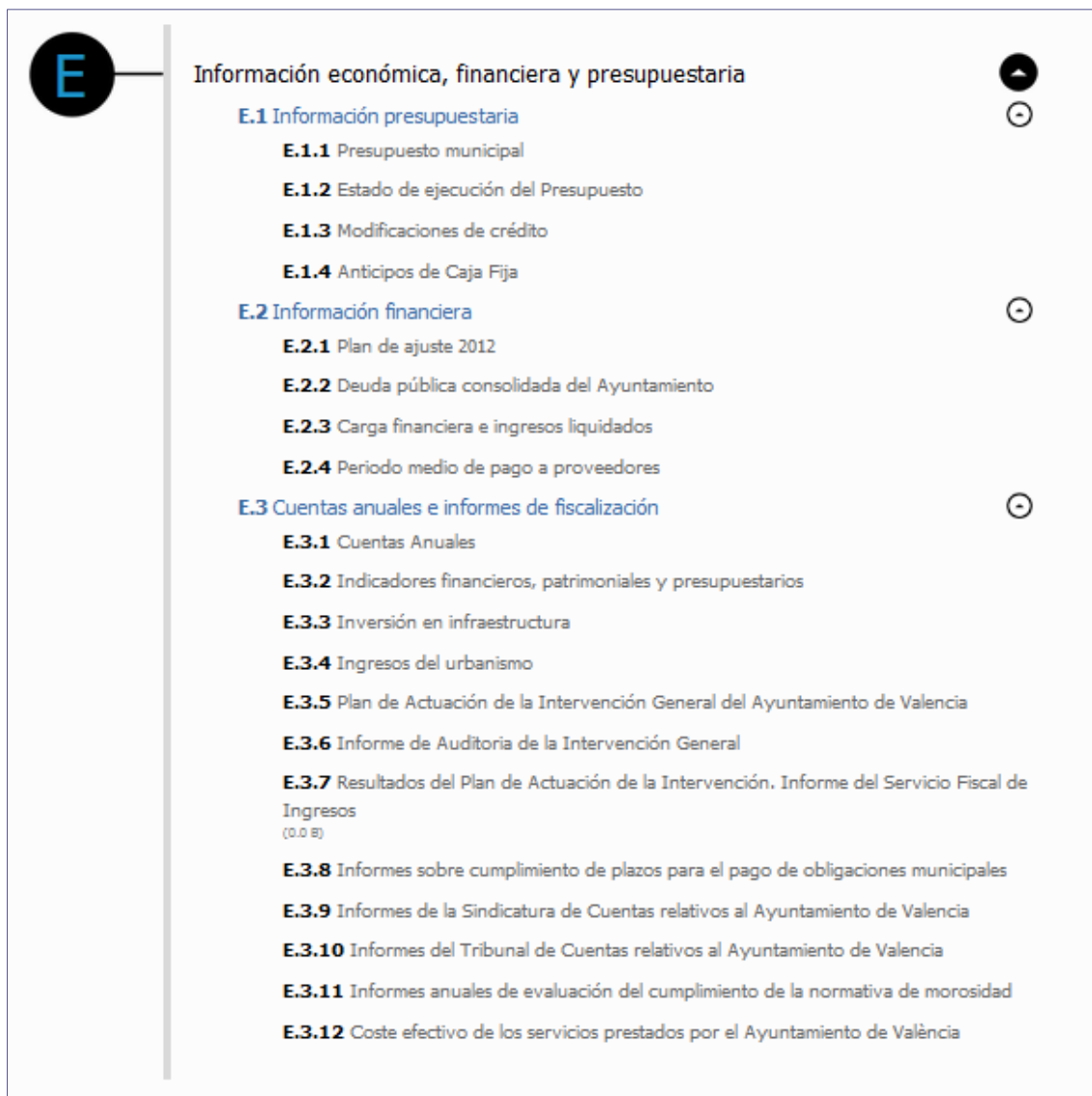


Figura 2.3. Captura apartado Transparencia.

Fuente: <http://gobiernoabierto.valencia.es/va/transparency/>. Último acceso: 1 agosto 2018.

En el apartado de datos abiertos hemos filtrado por los sectores “Hacienda” y “Economía”. Los *datasets* que encontramos se pueden ver en la Figura 2.4, solamente

2. Estado del arte

aparecen tres. El sector “Sector público” también lo consultamos, pero los *datasets* que nos mostraba, aparte de estos tres, no eran de nuestro interés.



Figura 2.4. Datasets económicos y financieros del portal de València.

Fuente: <http://gobiernoabierto.valencia.es/va/data/>. Último acceso: 1 agosto 2018.

Por último, vamos a revisar que podemos encontrar el visor presupuestario. Este apartado es de gran interés para nuestro proyecto, porque aparte de visualizar los datos presupuestarios como se muestra en la Figura 2.5, nos permite descargar la información en formatos reutilizables.

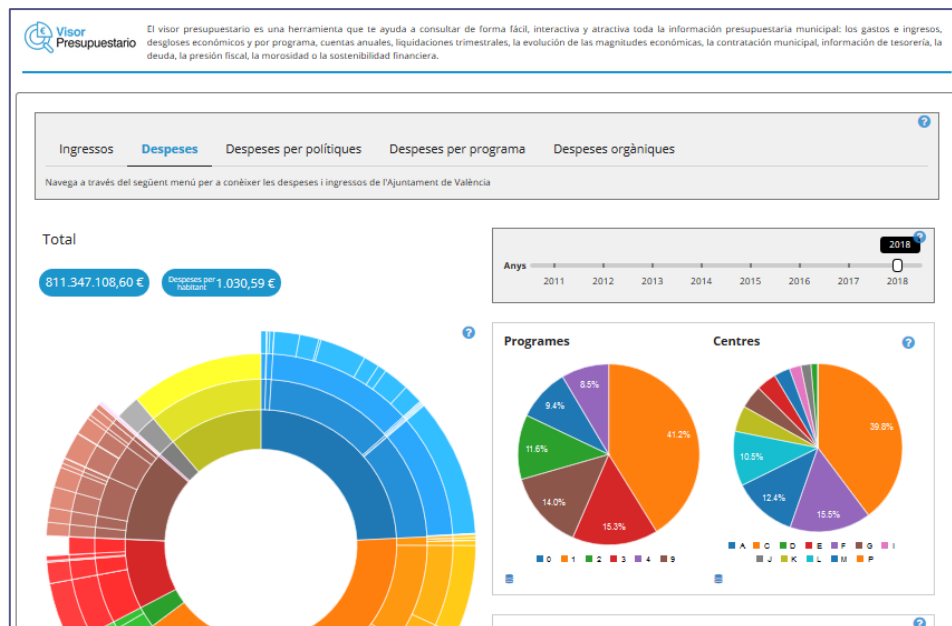


Figura 2.5. Visor presupuestario del Ajuntament de València.

Fuente: http://www.valencia.es/somclars/ca/#/?_k=p2po5w. Último acceso: 1 agosto 2018.

2.2 ASISTENTE VIRTUAL PARA ACCESO A INFORMACIÓN

En este apartado hablaremos de qué son los asistentes virtuales que tienen como función el acceso a la información. Para ello, primero realizaremos un repaso a qué son los asistentes virtuales de forma genérica y cuál es su estructura más habitual. Luego revisaremos qué son las aplicaciones de mensajería instantánea y cuál nos convendría utilizar. Para finalizar, analizaremos el estado del procesamiento del lenguaje natural (PLN) y ahondaremos en qué herramienta nos conviene más para hacer el PLN según las características de nuestro proyecto.

2.2.1 Asistentes virtuales

El término asistente virtual tiene distintos significados según el contexto, el primer significado se refiere a una profesión surgida con el boom de internet, sobre la década de los 90 que se encarga de prestar servicios de soporte administrativo, técnico, creativo, etc. de forma remota a través de internet. El otro significado, al que haremos referencia en este trabajo, surgió después del auge de la inteligencia artificial. Antes de definir qué es un asistente virtual, vamos a realizar un recorrido por los distintos términos que se utilizan como sinónimos. Lo haremos así porque no existe unanimidad en la terminología dentro del área de conocimiento y distintos autores se refieren con distintos términos y ligeras variaciones en el significado. Con este repaso homogeneizamos el término y lo definiremos para el presente trabajo.

En una página dedicada a los *chatbots* han llegado a recopilar 161 sinónimos de la palabra (Van Lun, s.f.), aunque es cierto que algunas definiciones son un poco atípicas. Los sinónimos más comunes de asistente virtual son: *chatbot* (tanto escrito junto como separado, *chat bot*), agente inteligente, agente conversacional, sistemas conversacionales y *chatterbot*. Entre todos los términos utilizamos asistente virtual debido a las funciones que va a desempeñar, ya que lo que se pretende es que sea capaz de interactuar con el usuario como si fuera una persona para tratar de solventar las necesidades de información del primero.

Entre las múltiples definiciones que podemos encontrar nos gustaría destacar dos. La primera publicada en una revista científica, es de las primeras definiciones que se pueden encontrar:

Un asistente virtual es un agente de software que realiza tareas automatizadas a través de una interacción humano - máquina a través de voz o texto (Hípola & Vargas-Quesada, 1999)

Y la segunda es una definición más moderna que incluye algunas novedades:

Es un programa de computadora que procesa la entrada de lenguaje natural de un usuario y genera respuestas inteligentes y relativas que luego se envían al usuario. (Khan & Das, 2017)

Como vemos en esta definición añade “respuesta inteligente” que corresponde al apartado de inteligencia artificial que permite que las respuestas sean más acertadas y que el comportamiento del asistente sea más parecido al de un ser humano. El otro aspecto diferenciador es que incorpora que la entrada es “lenguaje natural” en vez de textual o verbal, con esta diferenciación quiere decir que no solo hace referencia a la forma física de la entrada de la información, sino que hace referencia a la estructura lingüística en la que llega la información. Siendo procesable el lenguaje con la misma estructura hablada que la que usamos con otra persona. En resumen, la gran diferencia entre las dos definiciones es que entre ellas se desarrolló todas las tecnologías de aprendizaje automático permitiendo un gran avance en los asistentes virtuales. Visto lo cual asumimos la segunda definición, ya que se acerca mucho más a nuestro proyecto y es a la que nos ceñiremos en el resto del documento.

Ahora que ya tenemos claro qué es un asistente virtual, vamos a hacer un repaso de los elementos funcionales que debe tener. Un asistente debe contener al menos los siguientes 4 elementos (Rozga, 2018):

- **Integraciones de canales:** Es recomendable aislar todo lo que es la lógica del asistente de la integración con la plataforma de mensajería, de esta forma será muy fácil adaptar el asistente a cualquier otra plataforma.
- **Bot:** Un asistente virtual no deja de ser un *bot* complejo y este apartado se encarga de las solicitudes de los usuarios. En definitiva, es un servicio web que recibirá un mensaje del software de integración con alguna plataforma de mensajería en la que esté integrado y luego de hacer todos los procesos (lo que puede incluir comunicarse con otros componentes) responderá el mensaje.
- **Motor de comprensión del lenguaje natural:** El lenguaje humano escrito son datos no estructurados, aun así, necesitamos que el asistente virtual sea capaz de analizar esos textos y extraer de lo que está hablando el usuario. Esto se hace con técnicas de procesamiento del lenguaje natural ya sea implementando el proceso en código o utilizando herramientas que facilitan este proceso. En el siguiente capítulo podemos encontrar más detalle sobre este punto.
- **Motor de gestión del diálogo:** Se trata de un conjunto de reglas que permite modelar la conversación y regular el comportamiento del asistente. Estas reglas se ejecutan después del motor de comprensión del lenguaje natural, este nos habrá devuelto datos estructurados con la información que haya extraído de los datos no estructurados que entraron en él. Este motor es donde se encuentra realmente la lógica de negocio.



En la Figura 2.6 podemos observar las relaciones entre estos elementos, estos elementos se agruparán siguiendo los intereses del desarrollador, no teniendo por qué estar en módulos diferenciados del software. Pero por razones estratégicas se puede decidir unir elementos en el mismo módulo del programa, solo que si están separados el asistente será más adaptable a los cambios.

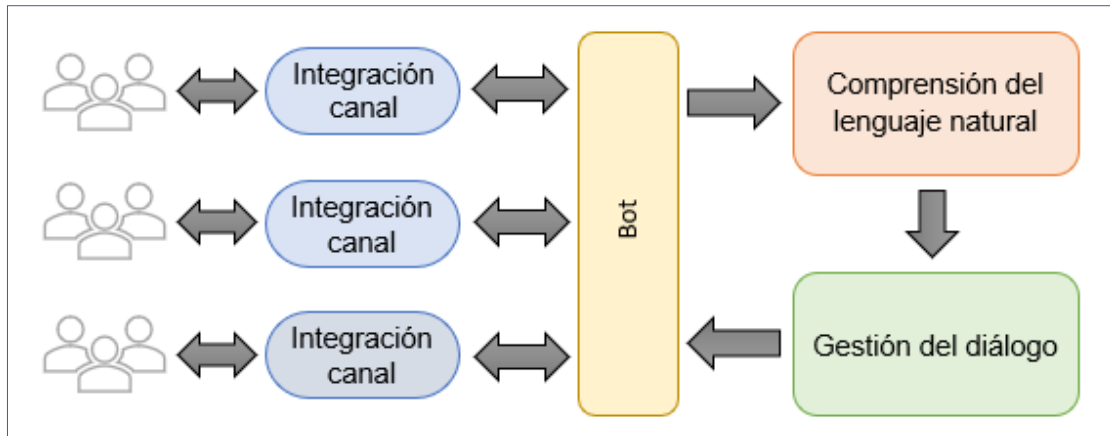


Figura 2.6. Relaciones elementos asistente virtual.
Fuente: Elaboración propia.

Otro aspecto para tener en cuenta es la evolución que han tenido los asistentes virtuales en su corta historia. Para ello haremos un repaso de los ejemplos más relevantes para lograr una mejor comprensión de la transformación que han sufrido. Uno de los primeros asistentes virtuales se llamaba ELIZA y fue creado por Joseph Weizenbaum en el MIT en 1966. ELIZA examinaba las palabras claves de las frases escritas por los usuarios y respondía en una frase modelo almacenada en su base de datos, de no tener respuesta utilizaba reglas de transformación para elaborar una respuesta a partir de la frase de entrada.

Las décadas posteriores a la creación de ELIZA el entorno de los asistentes virtuales siguió evolucionando con asistentes como Jabberwacky o A.L.I.C.E. Todos ellos seguían el mismo camino que ELIZA, pero no sería hasta el cambio de siglo que no llegará un verdadero salto en los asistentes virtuales con la llegada de SmarterChild, construido por ActiveBuddy el año 2001. Este fue el primer asistente que no solo tenía la función de entretener, sino que además era capaz de proporcionar información útil, y gracias a eso logró más de 30 millones de usuarios. Este asistente es considerado el padre de los asistentes virtuales modernos.

En la actualidad, la situación para los asistentes virtuales es muy positiva. La gran mayoría de grandes empresas tecnológicas pelea por tener el mejor asistente virtual en el mercado. Algunos ejemplos son: Siri de Apple, Google Assistant de Google, Cortana de Microsoft y Alexa de Amazon. Además de estos asistentes algunas plataformas como Telegram, Slack, Facebook Messenger o Skype han proporcionado soporte a la creación de *bots*. Esto unido a los avances de la tecnología, ha permitido democratizar la creación

tanto de *bots* como de asistentes virtuales ya que cualquier usuario con ciertos conocimientos puede desarrollar uno y compartirlo con el mundo (Khan & Das, 2017).

2.2.2 Mensajería instantánea

La mensajería instantánea es una forma de comunicación en tiempo real entre dos o más personas basada en texto entre dispositivos conectados a una red, normalmente internet. En la actualidad el uso mayor de mensajería instantánea se hace empleando aplicaciones de mensajería instantánea desde dispositivos móviles, aunque la mayoría de ellas tiene soporte web o aplicaciones que permiten utilizarlas desde ordenadores.

Estas aplicaciones de mensajería instantánea son la evolución natural de las antiguas salas de chat en las que mucha gente conversaba de forma escrita sobre algún interés en común. En la actualidad existen muchas aplicaciones de mensajería instantánea, cada una tiene sus características representativas e intenta aportar servicios y funciones que la distinga de sus competidoras. Entre las funciones que la gran mayoría incorpora aparte de permitir el intercambio de mensajes entre dos personas o en grupo están: intercambio de imágenes o archivos, llamadas de voz o videollamadas a través de internet o algunas funciones sociales.

Ahora, vamos a hacer un repaso a las cinco aplicaciones más utilizadas en España (CIS, 2016):

- **WhatsApp:** Aplicación desarrollada por WhatsApp Inc. y lanzada el 2009, posteriormente fue comprada por Facebook. Es la aplicación de mensajería instantánea más popular en España. En cuanto a las funciones han ido ampliándose, en la actualidad incluye llamada, videollamada y cifrado punto a punto para aumentar la privacidad de los mensajes.
- **Skype:** Aplicación lanzada el 2003 distribuido por Microsoft después de que lo comprara a Skype Technologies. Fundamentalmente es conocida por su función de hacer videoconferencias de hasta 10 personas, pero también incluye los servicios de mensajería.
- **Telegram:** Desde el año 2013 está disponible esta aplicación desarrollada por Telegram Messenger LLP. Aparte de hacer las funciones comunes de estas aplicaciones, esta aplicación hace una gran apuesta por la seguridad de las comunicaciones. La versión cliente tiene licencia abierta, por lo que existen distintas versiones cliente y permite mucha personalización. El último punto diferenciador es que permite crear grupos muy grandes, canales unidireccionales y la posibilidad de incluir *bots*.
- **Line:** En el año 2011 LINE Corporation sacó al mercado esta aplicación, con ella se pueden realizar, además de los servicios de mensajería instantánea, llamadas



de voz. Su rasgo más característico son la posibilidad de utilizar *stickers* en las conversaciones.

- **Facebook Messenger:** Originalmente a su publicación en 2008 se llamaba Facebook Chat, pero después de remodelarlo fue renombrado en 2011. Está desarrollada por Facebook Inc. la gran ventaja que tiene respecto el resto es que puedes conversar con todos los contactos que tengas en Facebook.

En la Tabla 2.1 recoge estas mismas aplicaciones destacando las algunas de las características que consideramos más relevantes a la hora de elegir una o otra plataforma.

	N.º Usuarios (año del dato)	Dispositivos⁸	Licencia	Soporte Bots
WhatsApp	Más de 1.000 millones (2017)	Android, iOS y PC	Software privativo	No
Skype	560 millones (2010)	Android, iOS y PC	Freeware	Sí
Telegram	200 millones (2018)	Android, iOS y PC	Servidor: Código cerrado Cliente: GPLv2 y GPLv3	Sí
Line	217 millones (2016)	Android, iOS y PC	Software privativo	Sí
Facebook Messenger	900 millones (2017)	Android, iOS y PC	Freeware	Sí

Tabla 2.1. Comparativa características mensajería instantánea.
Fuente: Elaboración propia.

En nuestro caso, en el marco de este trabajo tomamos la decisión de que entre las diferentes plataformas utilizaremos Telegram. Esto es porque, aunque nos hubiera encantado poder desarrollar el asistente virtual para la plataforma WhatsApp por la implantación casi total y la familiaridad que tiene la gente de todas las edades con esta plataforma, todavía no ofrece soporte para la utilización de *bots*. Por ello nos fuimos a una plataforma similar que sí los permitiera. Además, teníamos experiencia previa elaborando *bots* sencillos para Telegram lo que la hacía ideal para este proyecto más grande.

2.2.3 Procesamiento del lenguaje natural

El ser humano es un ser social, y el factor fundamental para que ello sea posible ha sido el lenguaje, en un principio de forma oral y posteriormente de forma escrita. Cuando hablamos de lenguaje natural hablamos de la comunicación hablada o escrita generada

⁸ Solo incluimos Android, iOS y si tienen soporte para PC

basada en una lengua humana, no serían lenguajes naturales los lenguajes construidos como lenguajes de programación u otros usados en la lógica formal.

El Procesamiento del Lenguaje Natural (PLN) (*Natural Language Processing (NLP)*, en inglés) “se concibe como el reconocimiento y utilización de la información expresada en lenguaje humano a través del uso de sistemas informáticos” (Sosa, 1997). Esta disciplina se sitúa dentro de la inteligencia artificial y desde los años 50 ha ido desarrollando sus técnicas con el objetivo principal de la comprensión del lenguaje humano por parte de la computadora y sus aplicaciones se centran en los siguientes aspectos:

- Traducción automática de textos. En el origen del PLN este era uno de sus principales objetivos.
- Procesamiento de textos. En la red existe multitud de información expresada de forma textual, en este ámbito de aplicación se desarrollan las tecnologías que permiten extracción de conocimiento de ellas. Algunos ejemplos serían la generación automática de resúmenes, el análisis de sentimiento, la detección de correos *spam*, etc.
- Interfaces en lenguaje natural. Estas aplicaciones se basan en establecer una comunicación hombre-máquina empleando lenguaje natural, en la que el usuario comunica las órdenes y directrices que debe ejecutar la máquina. Esta comunicación vendría a sustituir la interfaz gráfica de usuario clásica por una interfaz de usuario conversacional.

Después de ver las aplicaciones que puede tener el PNL, vamos a ver las dificultades a las que se enfrenta y los distintos niveles de procesamiento desde los que se abordarán para solucionarlo. La ambigüedad del lenguaje natural es el gran problema a la hora de intentar procesarlo, muchas veces hasta para otra persona es complicado interpretar algunas expresiones. Esta ambigüedad ya no son sólo las palabras con múltiples significados o la utilización de recursos literarios que puedan dificultar la comprensión, también entrarían en esta dificultad los errores ortográficos, regionalismos o expresiones irónicas o sarcásticas.

Para solventar estas ambigüedades hace falta un tratamiento riguroso a todos los niveles de análisis del lenguaje, normalmente se aborda desde los siguientes cuatro niveles: morfológico, sintáctico, semántico y pragmático. Además de estos puntos de análisis, en algunos casos para proyectos concretos se pueden considerar algunos niveles de conocimiento más, como por ejemplo el análisis de información fonológica si la entrada es sonora o de análisis del discurso que estudia el contexto comprobando cómo la información precedente puede ser relevante para la comprensión.

Para este proyecto nos vamos a centrar solamente en los cuatro niveles de análisis básicos. En la Figura 2.7 se ve el proceso de ejecución de los análisis, habitualmente es



secuencial y se ejecuta primero uno y el resultado de este es la entrada del siguiente. Ahora vamos a detallar cada uno de esos componentes o fases del proceso:

- **Análisis morfológico:** La unidad mínima que aporte significado de todo texto son las palabras. El análisis morfológico estudia la estructura y los mecanismos de formación de las palabras, en ocasiones también en llamado análisis léxico.
- **Análisis sintáctico:** Después de identificar y analizar las palabras que componen un texto, el siguiente paso consiste en analizar la organización y relación entre estas.
- **Análisis semántico:** El objetivo de este proceso es determinar de forma no ambigua el significado del texto, frase a frase.
- **Análisis pragmático:** En esta fase se busca información adicional al significado, obtenido en la fase anterior, en función del contexto.

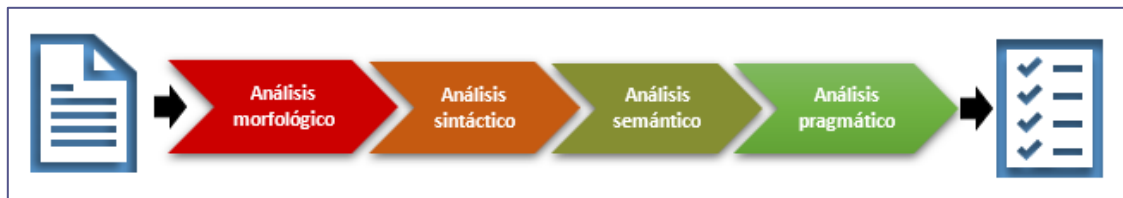


Figura 2.7. Fases de procesado del lenguaje natural.
Fuente: Elaboración propia

En resumen, primero la morfología extrae la información que se puede a nivel palabra, después el análisis sintáctico busca las relaciones entre estas palabras, luego la semántica ahonda el significado de las frases y el pragmático da una última revisión al significado de la frase según el contexto en el que está escrita.

Herramientas PLN para crear asistentes virtuales

Una vez aclarados los procesos que se realizan para procesar el lenguaje natural vamos a explorar las herramientas disponibles que nos permitan agilizar el desarrollo de nuestro asistente virtual. En los últimos años, se han incrementado las herramientas y servicios que facilitan la creación de todo tipo de asistentes virtuales. En este apartado vamos a hablar de algunas de las herramientas más potentes que hacen la función del motor de comprensión del lenguaje natural y nos ofrecen un motor de gestión del diálogo.

Las principales empresas tecnológicas ya tienen sus herramientas PNL: Dialogflow de Google, LUIS de Microsoft, Watson de IBM y Wit.ai de Facebook. En este apartado exploraremos un poco de cada una y haremos una comparativa para ver cuál nos conviene utilizar, pero antes de ello vamos a explicar unos conceptos básicos que son comunes en el funcionamiento de todas.

Todas las herramientas PLN para crear asistentes virtuales utilizan esquemas similares y utilizan conceptos en común relativos a las interfaces de usuario conversacionales. El concepto central de cualquier interfaz de usuario conversacional es el **Intent**, es decir la intención del texto. Lo primero que se debe hacer cuando se reciba un mensaje textual es asociar la frase a una acción específica de las que podemos proporcionar. Por lo cual deberemos tener un **Intent** por cada acción que realice el asistente, pero esa acción se puede activar con frases múltiples.

Junto a los **Intent**, el otro concepto esencial es la **Entity**, estas entidades tienen la función de extraer los parámetros necesarios para llevar a cabo el **Intent** que incluya la frase. Las herramientas habitualmente tienen entidades predefinidas para conceptos comunes como expresar fecha y hora, números, localizaciones, etc.

Llegados a este punto sólo falta establecer el significado de sesión y contexto. Una **sesión** es una conversación de principio a fin en un tiempo determinado, desde el primer texto, pasando por todos los **Intents** que sean necesarios a través de interacciones preguntas y respuestas, hasta que se da por terminada esa interacción. En cambio, el **contexto** almacena los estados intermedios y los parámetros de expresiones anteriores dentro de una sesión. Puede haber múltiples contextos anidados, pero solo podrá haber una sesión por usuario a la vez.

En la Figura 2.8 podemos ver una interfaz conversacional, un ejemplo de conversación en el cual un usuario está intentando reservar un vuelo a través de un asistente virtual y a modo de resumen de los anteriores conceptos los podemos ver identificados.

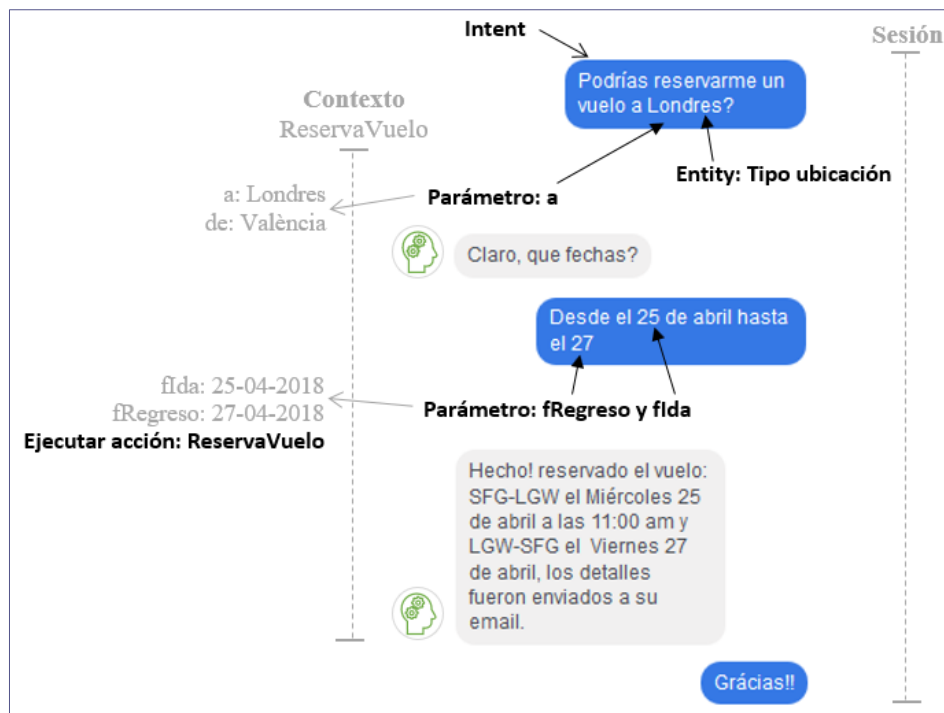


Figura 2.8. Conversación simulada de bot con aclaración de conceptos.

Fuente: Elaborado a partir de <https://stanfy.com/blog/advanced-natural-language-processing-tools-for-bot-makers/>. Último acceso: 1 agosto 2018.

A continuación, vemos las herramientas PLN más importantes para la creación de asistentes virtuales:

Dialogflow

Plataforma lanzada en 2010 con el nombre de api.ai. En 2016 fue adquirida por Google y renombrada como Dialogflow. Permite usar *Intents*, *Entities* y contextos. Aunque originalmente era gratuita, después de la compra por parte de Google cambiaron las políticas y ahora existen dos tipos de cuentas, la versión gratuita, que permite desarrollar asistentes limitando el número de interacciones por hora, y la versión para empresas, que pagan por el uso.

Los puntos fuertes de Dialogflow son que permite el despliegue en casi cualquier plataforma de mensajería instantánea, incluye herramientas de monitorización y permite integración con otros sistemas (por ejemplo, bases de datos) empleando servicios web (*webhook*). Además, existe buena documentación y ejemplos para facilitar la entrada. En la parte negativa podemos destacar que no dispone de una visualización gráfica de los flujos de conversación como tienen otras.

Wit.ai

Creado en 2013 y posteriormente comprada por Facebook el 2015. Además de usar *Intents* y *Entities* utiliza *Stories* (historias) es una forma más intuitiva de acercarse al contexto de la conversación. En vez de crearse *Intents* solos, se crean flujos de conversación que incluyen varios *Intents* que se abren a condición de lo que diga el usuario.

Permite monitorizar los asistentes con Facebook Analytics y cuenta con buena documentación. Su punto más fuerte es las *Stories*, que permiten desarrollar asistentes sencillos con poco esfuerzo. En su contra no está preparado para integrar sistemas mediante servicios web.

LUIS (Language Understanding Intelligent Service)

Plataforma de Microsoft publicada en 2015. Es una de las soluciones más completas, utiliza de forma similar a Dialogflow los *Intents* y las *Entities*. no dispone de un creador gráfico de conversaciones mediante flujos. Sus puntos fuertes son el conjunto de servicios que también proporciona Microsoft para ayudar a la creación de asistentes y las métricas son las más valiosas para comprender cómo está funcionando el asistente.

Como punto negativo es el precio por uso. Además, al estar integrado en Microsoft Azure y facilitar la interacción entre todos los servicios adicionales puede incrementar bastante el precio de desarrollo.

Watson

Publicado en 2016 por parte de IBM. Está basado en el uso de *Intents*, *Entities* y contextos, pero además de eso añade funcionalidades para el reconocimiento de sinónimos y otro de corrección de palabras mal escritas. También permite crear flujos de diálogos de forma gráfica. Igualmente, permite desplegar los asistentes en algunas herramientas como Slack, Facebook Messenger y en la web.

El punto fuerte de Watson es el ecosistema de APIs que añaden muchas facilidades como convertir audio en texto, traducir textos o reconocimiento de imágenes. Además, incluye herramientas de monitorización y está bien documentada. Tiene el inconveniente que la parte gratuita es muy limitada y el resto es de pago.

Comparativa

Para facilitar la elección se ha elaborado la Tabla 2.2 para resumir los aspectos importantes de las cuatro herramientas, teniendo en cuenta las características de nuestro proyecto y evitando las características que todas cumplen como son la inclusión de *Intents*, *Entities* y contextos de alguna forma o la monitorización de los asistentes.

	Dialogflow	Wit.ai	LUIS	Watson
Idioma Español	Sí	Sí	Sí	Sí
Idioma Catalán	No	Sí	No	No
Gratuito	Sí, limitado	Sí	Sí, limitado	Sí, limitado
Integración Telegram	Sí	No directamente	Sí	No directamente
Visualización flujos de conversación	No	Sí	No	Sí
Integración servicios web personalizados	Sí	No	Sí, limitado a Azure	Sí

Tabla 2.2. Comparativa de las características herramientas PLN.
Fuente: Elaboración propia.

Después de todo lo visto en los puntos anteriores y la tabla resumen, apostamos por desarrollarlo en Dialogflow porque se adapta mejor a las características que necesitamos para nuestro proyecto: tiene soporte para español, la limitación de la versión gratuita no nos afecta demasiado ya que para desarrollar un prototipo nunca llegaremos a esos ratios, nos permite la integración en Telegram de forma muy sencilla y la conexión a un servicio web donde poder consultar las respuestas de nuestro asistente. En contra tenemos que no permite diseñar flujos de conversación de forma gráfica, pero podemos paliar esto haciendo un correcto uso de los contextos. A su vez, el otro inconveniente es que no tiene soporte en catalán, de querer que el asistente sea bilingüe podría llevarnos más trabajo, pero en la mayoría no está incluido así que no es un factor determinante para hacernos elegir otro.

3 PLANTEAMIENTO DE LA SOLUCIÓN

En primer lugar, se hace un análisis detallado del problema. A continuación, se plantea la solución, y finalmente se explica la metodología de desarrollo propuesta para la implementación de la solución planteada.

3.1 ANÁLISIS DEL PROBLEMA

El problema que vamos a abordar es la desconexión de la ciudadanía con los datos económicos y financieros del Ajuntament de València. Revisando las estadísticas de acceso a la información procedente del portal de transparencia y datos abiertos del Ajuntament de València, se aprecia que las cifras de acceso son muy bajas.

En la sección “Catálogo de datos” del citado portal se muestra el número de visitas que ha recibido un *dataset* desde su creación. En materia económica y financiera solo hay publicados 3 *datasets*, de los cuales el más visitado no supera las 1300 visitas. Por otro lado, la sección de transparencia y el visor presupuestario no aportan los datos del número de visitas ni de descargas. Así que no podemos conocer si la ciudadanía está consultando en estos apartados la información económica y financiera. Con esta ausencia de información sólo podemos extrapolar y teorizar que las visualizaciones del resto de información de la misma temática deben ser similares.

A partir de este primer acercamiento se puede detectar otro problema, solo en el portal de transparencia y datos abiertos hay tres apartados donde encontrar información en materia económica. Esta dispersión puede ser una de las causas posibles de por qué los números de visualización son tan bajos. Cuando un ciudadano se enfrenta a buscar información en el portal, no sabe por dónde comenzar y en muchas ocasiones termina desistiendo antes de cumplir el objetivo.

Pero no solamente la dispersión de la información puede ser la causa de la falta de presencia ciudadana en estos portales, también afectan factores como la publicación de información en formatos complicados de tratar para la mayoría de la ciudadanía y el uso de un lenguaje técnico que es ajeno al común de los ciudadanos.

Existen muchas más causas posibles, pero para este proyecto nos centraremos en resolver los que hemos identificados.

3.2 PLANTEAMIENTO DE LA SOLUCIÓN

Para dar solución al problema, se decide desarrollar un asistente virtual integrado en la aplicación de mensajería instantánea Telegram que permite la consulta de información económica y financiera del Ajuntament de València.

De esta forma, cualquier persona interesada o simplemente con curiosidad en algún aspecto de esta información, podrá consultarla a través de Telegram; simplemente manteniendo una conversación con el asistente en lenguaje natural como si estuviera chateando con cualquier otro contacto. Con la ventaja de no tener que instalar otra aplicación específica en su dispositivo móvil y no tener la necesidad de acceder a un portal web.

Debido a que la información se encontraba dispersa, se consideró necesario la creación de una base de datos intermedia que agrupara toda la información necesaria para responder las preguntas. Esto nos aporta dos beneficios:

- Control del histórico de los datos: aunque el ayuntamiento elimine los datos al actualizar los *datasets*, estos seguirán disponibles para el asistente.
- Exportabilidad: al publicar la estructura definida, cualquier entidad que desee utilizar este asistente, solo necesitará almacenar sus datos siguiendo esta estructura y conectar el resto del asistente para hacerlo funcionar.

Para realizar la gestión del lenguaje natural se utiliza la herramienta Dialogflow y de forma separada se confeccionará el gestor de respuestas que será la que se comunique con la base de datos. En el punto de integración con Telegram nombraremos al asistente cómo **infoecoVLC**, el nombre surge de las abreviaciones de información económica y las siglas de València. Con esto no queremos decir que desarrollaremos una aplicación a medida para este ayuntamiento, sino que se pretende crear una solución que pueda ser exportada fácilmente a cualquier otro ayuntamiento que le interese sacar provecho de sus datos. Por tal motivo, si otro ayuntamiento decide implantar el asistente se nombraría infoecoXXX siendo las XXX las siglas de la ciudad. Para garantizar la exportabilidad, toda la documentación y código que se produzca en el desarrollo de la solución será puesta a disposición del público con licencias abiertas.

Es conveniente destacar que, en el marco de este proyecto no desarrollaremos la solución final, sino que desarrollaremos un prototipo funcional con el objetivo de probar el flujo de interacción, herramientas acordes a este tipo de desarrollo y redactar los requerimientos formales con vistas a la solución final.

3.3 METODOLOGÍA

La metodología seleccionada para el desarrollo del proyecto seguirá un modelo de prototipado evolutivo. Se trata de una metodología realmente útil para realizar aplicaciones donde se tienen claros los objetivos generales para el desarrollo del software, pero donde los requerimientos están aún por definirse, como es nuestro caso.

Al tratarse de prototipado evolutivo, el resultado de cada iteración será la entrada de la siguiente, generando como resultado en cada iteración, una versión funcional que progresivamente irá mejorando. De esta manera, tenemos un prototipo funcional desde la primera iteración.

En este tipo de metodologías es frecuente que el cliente defina un conjunto de objetivos generales para el software, pero en nuestro caso no contamos con un cliente de forma tradicional, sino que tenemos por cliente a la ciudadanía. Por ello, deberemos adaptar la metodología a estas necesidades, a través de la integración de técnicas de diseño centrado en la ciudadanía. Esta integración se realiza en las fases de comunicación y evaluación y *feedback* con el objetivo de definir los requisitos. Estos pequeños cambios permitirán que el resultado final esté más centrado en las necesidades de los ciudadanos, ya que adaptamos a la solución los nuevos requerimientos que vayan apareciendo fruto de la interacción con la ciudadanía.

En la Figura 3.1 se pueden observar las etapas que seguiremos en la metodología, estas serán: comunicación, planificación, modelado, construcción y evaluación y *feedback*.

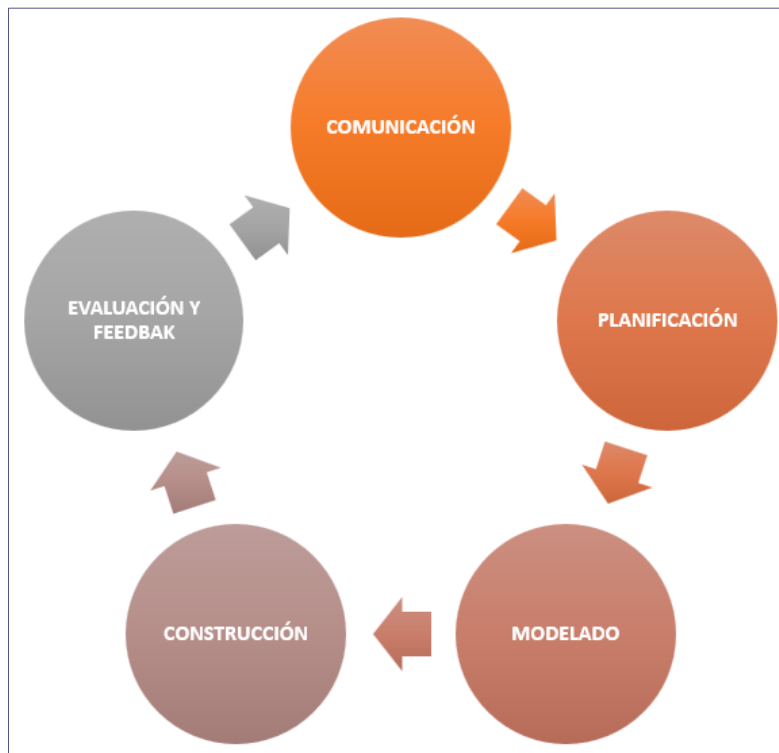


Figura 3.1. Etapas modelo de prototipado evolutivo.
Fuente: Elaboración propia.

- Etapa de **comunicación**, en esta etapa se debe recopilar la información que nos permitirá definir los requisitos en la etapa de planificación. Para recopilar esta información se debe establecer un diálogo con la ciudadanía, mediante encuentros presenciales, así como encuestas en línea.
- Etapa de **planificación**, en esta etapa se realiza la definición formal de los requerimientos a partir de los resultados de la fase anterior. Los requerimientos se dividen en funcionales y no funcionales:
 - Los requerimientos funcionales permiten determinar qué hará y qué no hará el software a partir de las relaciones de su operación y su implementación, así como las validaciones que se deben realizar para determinar el comportamiento que debe tener el sistema.
 - En cuanto a los requerimientos no funcionales, se enfocan en las restricciones de los servicios o funciones ofrecidos por el sistema, como son, el tiempo, el proceso, la usabilidad, portabilidad, etc.
- Etapa de **modelado**, en esta etapa se llevan a cabo todas las tareas correspondientes al diseño de la solución, incluida la arquitectura y la identificación de las tecnologías que se verán implicadas. Hay que resaltar, que es aquí donde se deben identificar las fuentes de datos necesarias para construir la respuesta que debe proporcionar el asistente, siempre y cuando en los requerimientos se haya determinado que se debe insertar un nuevo *Intent* al asistente.
- Etapa de **construcción**, en esta etapa se realizará la construcción del prototipo, una vez establecido en la etapa anterior la arquitectura y las tecnologías implicadas. En este momento, se utiliza el lenguaje de programación seleccionado para desarrollar el código fuente y las herramientas seleccionadas se configuran para que manejen parte de las funcionalidades del asistente.
- Etapa de **evaluación y feedback**, en esta etapa el usuario evalúa el prototipo, lo que permite valorar si se cumplieron con los objetivos, a la vez que aprendemos lo que desea el usuario con vistas al producto final. Esta evaluación se realizará empleando técnicas de diseño centrado en la ciudadanía, en nuestro caso haciendo pruebas de usabilidad.

Una vez terminadas todas las etapas se valorará si el estado del prototipo es el adecuado para ser considerado el prototipo final o si por el contrario deberá volver a iniciar las etapas para incorporar nuevos requisitos.



4 DESARROLLO DE LA SOLUCIÓN

En el capítulo anterior hemos explicado la metodología de desarrollo propuesta para la implementación de la solución planteada, siguiendo el modelo de prototipado evolutivo. En este capítulo seguimos el esquema iterativo, donde el número de iteraciones está dado por el tiempo que se ha dispuesto para el desarrollo del prototipo. En cada iteración se añade una cuestión, entendiendo cuestión como un *Intent*⁹, es decir, una acción de consulta de información al asistente que se activa con múltiples preguntas.

Para el desarrollo de la solución contamos con la colaboración de otro miembro de la Càtedra Govern Obert que durante todas estas iteraciones nos ayudó en la identificación y recopilación de las fuentes de información y, se encargó de la construcción de los ETL.

Antes de comenzar con la descripción del desarrollo de la solución, resulta necesario realizar un inciso sobre la gestión del código fuente del proyecto. Para el control de versiones utilizamos GitHub¹⁰ y lo publicamos en esta plataforma empleando la licencia abierta “GNU General Public License v3.0”¹¹. De esta forma, todo el conocimiento que generemos podrá ser retomado por cualquier persona u organización y, ampliado o adaptado a su realidad.

4.1 ITERACIÓN 1

Esta es la primera iteración del desarrollo, siendo la más extensa de todas porque en ella definimos la arquitectura de nuestra aplicación. Esto hace que esta iteración se extienda más en el tiempo, pero traerá la ventaja de que al detenernos a reflexionar sobre la arquitectura, se sienten las bases para que en las demás iteraciones se incluyan los *Intents* de forma más fácil, estableciendo una estructura común que acepte todo tipo de pregunta.

Comunicación

Para tener claro el público objetivo de nuestra aplicación, lo primero que decidimos hacer fue establecer el perfil de usuario, que permitiera en posteriores fases redactar mejor los requerimientos. Con respecto a los usuarios, se estableció que:

⁹ En el apartado 2.2.3 puede encontrar información más detallada del significado de *Intent*

¹⁰ Todo el código se puede encontrar en: <https://github.com/arnauc6/infoecoVLC>. Último acceso: 14 septiembre 2018.

¹¹ Detalle de la licencia en: <https://www.gnu.org/licenses/gpl-3.0.en.html>. Último acceso: 04 septiembre 2018.

Los potenciales usuarios de nuestro asistente son hombres y mujeres que viven en el área metropolitana de la ciudad de València. Mayores de 16 años con dispositivos móviles propios y usuarios habituales de programas de mensajería instantánea e interesados en la política, en general; y, en la política de ámbito municipal, en particular.

En esta fase también se llevó a cabo una pequeña encuesta con el fin de orientar mejor el proyecto. La encuesta se realizó a un grupo de personas reunidas con motivo de una prueba de usabilidad efectuada por la Càtedra Govern Obert para el Ajuntament de València. Para esta prueba se reunió un grupo de 24 personas, de las cuales 8 pertenecían a asociaciones ciudadanas y 16 a distintos departamentos del Ajuntament de València, con el fin de evaluar la herramienta que permite visualizar los presupuestos que en ese momento iba a implementar el ayuntamiento en el Portal de transparencia y datos abiertos¹².

Al finalizar la citada prueba, pasamos a los asistentes las cinco preguntas que serían de interés para nuestro trabajo. El cuestionario pretendía recabar información sobre la utilidad que representaría para ellos la existencia de otro mecanismo de contacto con el ayuntamiento, si incorporaría un sistema de búsqueda de contenidos, qué preguntas le harían a un asistente virtual y si valorarían positivamente acceder a la información económica y presupuestaria del ayuntamiento desde dispositivos móviles¹³.

Estas preguntas permitieron confirmar la necesidad de añadir nuevas formas de búsqueda e incluso el uso de lenguaje natural o lenguaje comprensible para los ciudadanos a la hora de realizarlas. Los encuestados mostraron interés en acceder desde los teléfonos móviles a la información y hacer preguntas sobre gastos concretos y gastos en los barrios.

En definitiva, la solución que planteamos está en sintonía con las respuestas que nos proporcionaron. En lo que respecta a las preguntas que debemos incluir a nuestro asistente virtual, debemos intentar abarcar preguntas más concretas porque suscitan más interés que las preguntas generales, aunque muchos de los datos necesarios para responder estas preguntas actualmente no se encuentran publicados por el ayuntamiento.

Planificación

A partir de la información obtenida en la fase anterior, realizamos la siguiente definición de requerimientos funcionales y no funcionales.

¹² Más información respecto al visor presupuestario en el apartado 2.1.4.

¹³ En el Apéndice A puede ver toda la información respecto a esta prueba.

Requisitos Funcionales:

Nombre	Responder a saludos.
Descripción: El sistema debe responder a frases de saludo y a mensajes de inicio de <i>bot</i> de la plataforma Telegram.	
Precondiciones: <ul style="list-style-type: none"> • El usuario debe introducir un texto que pueda ser interpretado como saludo por el agente 	
Flujo Normal: <ol style="list-style-type: none"> 1. El usuario manda un mensaje vía Telegram. 2. El agente recibe el mensaje. 3. El agente decide que se trata de un saludo. 4. Se responde con una de las respuestas tipo. 	
Flujo Alternativo: <ul style="list-style-type: none"> • 	
Postcondiciones: <ul style="list-style-type: none"> • 	

Nombre	Responder un mensaje tipo de error.
Descripción: El sistema debe responder un mensaje tipo de error si no corresponde a las preguntas válidas.	
Precondiciones: <ul style="list-style-type: none"> • El usuario debe mandar una pregunta que no esté contemplada en ningún <i>Intent</i>. 	
Flujo Normal: <ol style="list-style-type: none"> 1. El usuario manda un mensaje por alguna de las entradas admitidas. 2. El agente recibe el mensaje. 3. El agente comprueba si se trata de alguna pregunta. 4. No se parece a ninguna pregunta almacenada. 5. Se responde con una de las respuestas tipo. 	
Flujo Alternativo: <ol style="list-style-type: none"> 4. Se parece a alguna pregunta almacenada. 5. Se responde con la respuesta definida en el sistema para esa pregunta. 	
Postcondiciones: <ul style="list-style-type: none"> • La pregunta queda almacenada para futuro tratamiento. 	

Nombre	Responder a preguntas sobre cuánto se paga de impuestos por barrios.
Descripción: El sistema debe responder a preguntas sobre cuánto se paga de impuestos por barrios. Solo responde Totales, no otras operaciones.	
Precondiciones: <ul style="list-style-type: none"> • La entrada del parámetro barrio es textual. • El nombre del barrio y el nombre del impuesto deberá coincidir con alguno de los sinónimos definidos. 	
Flujo Normal: <ol style="list-style-type: none"> 1. El usuario manda un mensaje por alguna de las entradas admitidas. 2. El agente recibe el mensaje. 3. El agente decide que se trata de este tipo de pregunta. 4. El agente hace una petición al servicio web en busca de la respuesta. 5. El servicio web encuentra la respuesta. 6. El servicio web manda la respuesta al agente y responde la pregunta. 	
Flujo Alterno: <ol style="list-style-type: none"> 5. El servicio web tarda en responder. 6. Se responde con un mensaje de error. 	
Postcondiciones: <ul style="list-style-type: none"> • Si el nombre del barrio se solicitará nuevamente el dato. • Si el nombre del impuesto no coincide se asume que está solicitando el valor total. 	

Requisitos No Funcionales:

- El sistema solo debe aceptar entradas en formato texto. Por ahora nos limitamos a que el asistente entienda los textos que se le envían (descartando imágenes o videos), por ser la base de comunicación con un asistente virtual y el único elemento necesario estimado por nosotros para que haya una comunicación.
- El sistema debe aceptar entradas desde Telegram, ya que hemos decidido que el asistente estará publicado en dicha plataforma.
- El sistema debe ser bilingüe, debe aceptar entradas y proporcionar respuestas tanto en valenciano como en castellano. Dado que entre los potenciales usuarios habrá grupos de hablantes de los dos idiomas, tenemos que estar preparados tanto para entenderlos como para responderles en el idioma que se sientan más cómodos.

- La gestión de respuestas debe realizarse en un servidor web separado del motor que interpreta el lenguaje natural. De esta forma, serán independientes e intercambiables individualmente si encontramos herramientas o mejores formas para realizar las funciones.
- El sistema almacenará todos los mensajes que reciba el asistente virtual. Al tratarse de prototipos esta información nos permitirá llevar un seguimiento para comprobar el funcionamiento.
- El sistema guardará el ID del usuario con el nombre e idioma de comunicación. De esta forma, seremos capaces de responder en el idioma que quiere el usuario sin tener que preguntarle cada vez que se conecte. Además, podremos llevar un control del número de usuarios únicos que se han puesto en contacto con nosotros.
- El sistema podrá desactivarse cambiando una variable (Activo / Inactivo). De esta forma, el desarrollador podrá desactivar el asistente para tareas de mantenimiento o cuando por alguna razón sea necesario dejar fuera del aire al asistente.

Modelado

Teniendo claro los requisitos, en este apartado vamos a definir la arquitectura del asistente virtual siguiendo una estructura modular. En cada módulo estableceremos el diagrama de procesos y las comunicaciones entre los diferentes módulos.

Para definir la arquitectura tomamos como base la estructura de los elementos que componen un asistente virtual mostrado en la “Figura 2.6. Relaciones elementos asistente virtual.” del capítulo 2, pero con algunas modificaciones, tal y como se muestra en la Figura 4.1. Dado que en estos momentos no tenemos pensado llevar el asistente a otras plataformas, unimos la integración del canal Telegram con el motor del *bot*, es decir, la integración de Telegram se encargará de dirigir todas las comunicaciones del usuario a los módulos de procesamiento del lenguaje natural y de gestión del diálogo. Al módulo de procesamiento del lenguaje natural le enviaremos el texto del mensaje creado por el usuario, para que nos retorne un JSON con la clasificación del *Intent* y con las *Entity* extraídas. Luego, si es pertinente, esta respuesta será enviada al servicio web que hace la función de gestor del diálogo para leer el *Intent* y las *Entity* y realizar la consulta en la base de datos para elaborar la respuesta.

El último módulo, encargado de la Extracción, Transformación y Carga, no es propiamente parte del asistente virtual, aunque es necesario para el funcionamiento de todo el sistema. Este es el punto en el que se establecen los procesos que finalizan con el almacenamiento de la información estructurada de forma tal, que pueda ser usada sin problemas por el servicio web.

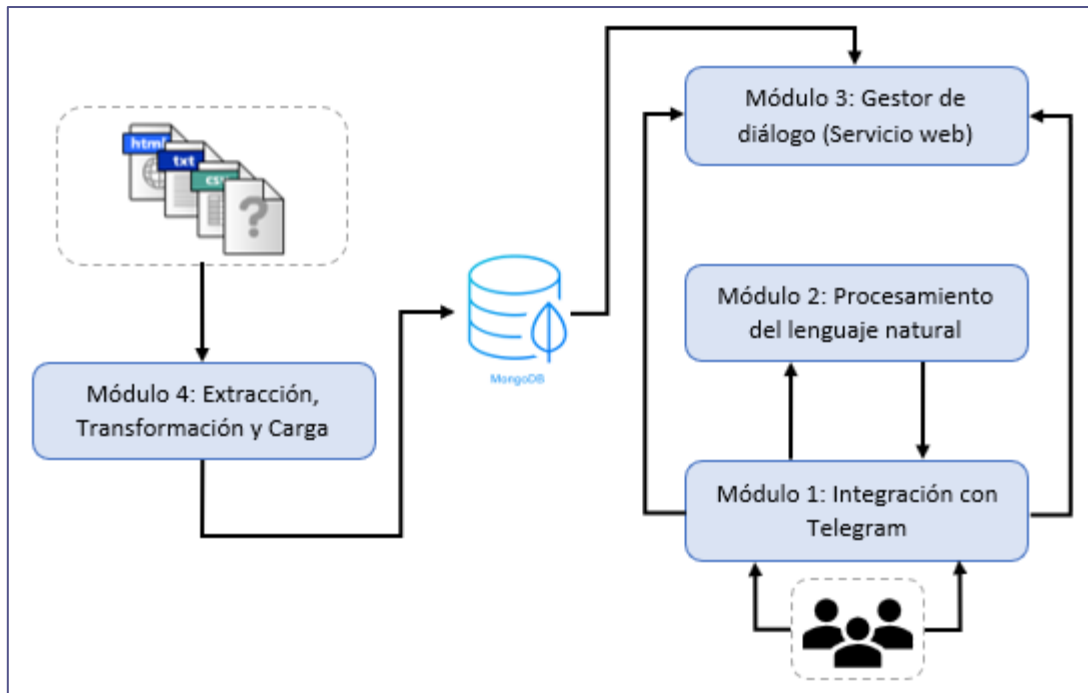


Figura 4.1. Diagrama de módulos del asistente virtual.
Fuente: Elaboración propia.

Las tecnologías que se ven implicadas en este proyecto son: Python para el módulo 1, 2 y 4; donde cada módulo hará uso de las librerías acorde con las funciones que desarrolla. Elegimos Python porque, aparte de conocer este lenguaje de programación, éste dispone de una gran comunidad que realiza desarrollo de aplicaciones de ciencias de datos y cuenta con multitud de librerías que permiten realizar todas las tareas que son necesarias.

Para la base de datos seleccionamos MongoDB, la base de datos NoSQL almacena todos los registros siguiendo la estructura del formato JSON, en vez de tablas. Esto supone una ventaja a la hora de comunicarse con nuestra aplicación, ya que todas las comunicaciones se realizan en este formato: el envío de mensajes de Telegram al asistente, la comunicación con Dialogflow y la comunicación con el servicio web. Adicionalmente, al no tener una estructura relacional nos permite ampliarla en cada iteración, expandiendo la base de datos poco a poco sin perder eficiencia en las consultas como nos pasaría con SQL, si no hiciéramos un buen diseño desde el principio.

El módulo de procesamiento del lenguaje natural lo delegamos en la herramienta Dialogflow, como justificamos en el punto 2.2.3, y nos encargaremos de la gestión y configuración para que funcione correctamente.

Después de explicar la arquitectura del asistente virtual vamos a repasar módulo a módulo el funcionamiento y los diagramas de proceso si procede:

Módulo 1: Integración con Telegram

El presente módulo se encarga de realizar toda la gestión interna de los mensajes recibidos. Cuenta con un mecanismo que se encarga de filtrar los mensajes, dejando pasar sólo aquellos que vienen en formato texto. Además, se encarga de gestionar las respuestas a los usuarios, realizando el direccionamiento de los mensajes con los módulos de procesamiento del lenguaje natural y el gestor de diálogo, hasta que se envía la respuesta al usuario.

En la Figura 4.2, podemos observar el diagrama de procesos de la integración en Telegram. Dentro de estos procesos, el sistema se encuentra a la espera de recibir un mensaje mediante el API de Telegram. Una vez recibido, se almacena en la base de datos sin realizar ningún tratamiento previo, con toda la información que nos envía Telegram¹⁴. Por lo tanto, si se produce un error podemos identificar el mensaje que lo provocó y buscar una solución con conocimiento de causa.

Cada pregunta que realizan los usuarios nos sirve para conocer sobre qué temas están interesados y poder crear en iteraciones futuras, nuevos que se ajusten mejor a sus intereses.

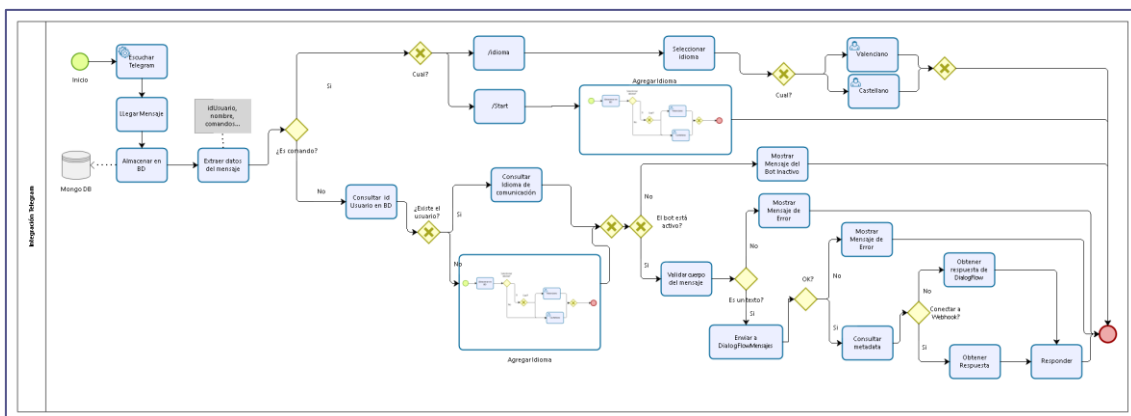


Figura 4.2. Diagrama de procesos de la integración en Telegram.
Fuente: Elaboración propia.

Una vez almacenado el mensaje, extraemos los datos de los mensajes que son relevantes para el futuro tratamiento: el id del usuario, el nombre que usa en Telegram y una entidad que nos aporta Telegram que nos indica si el mensaje es un comando o no. Si el mensaje recibido es un comando, identificamos si corresponden con los que tenemos disponibles: `/start` y `/idioma`. Que realizan lo siguiente:

- `/start`: se envía automáticamente cuando el usuario entra por primera vez. Se responde con un saludo y se le pide el idioma con el que quieres que se

¹⁴ Más información sobre la estructura de los mensajes: <https://core.telegram.org/bots/api#message>.
Último acceso: 1 septiembre 2018.

comunique el asistente. Si no responde, por defecto tomamos el castellano como idioma de comunicación.

- */idioma*: Se puede emplear en cualquier momento para modificar el idioma de las comunicaciones.

De no ser un comando, consultamos en la base de datos si es un usuario conocido, es decir, si tenemos sus datos almacenados. De no serlo, seguimos el mismo camino que el comando */start*. En caso de que el usuario sí esté registrado en la base de datos, consultamos el idioma de comunicación y lo agregamos al resto de datos del mensaje y a partir de este punto el asistente siempre reconozca el idioma con el que debe responder.

Una vez el sistema tiene identificado usuario y el idioma de las comunicaciones, realiza las comprobaciones de si el asistente está activo o en mantenimiento y si la entrada es textual. Para seguir, debe estar el asistente activo y la entrada debe ser textual, si no es así, se responden mensajes de alerta. Por la relevancia que tiene el fragmento siguiente del diagrama y para que se aprecie mejor las comunicaciones con los otros módulos, en la Figura 4.3 podemos encontrar el detalle del diagrama.

Llegados a este punto, solo queda mandar a la API¹⁵ de Dialogflow el texto introducido por el usuario y según la respuesta obtenida responder directamente o realizar la llamada al gestor de diálogos para buscar la respuesta. En cualquiera de los dos casos, terminan mandando al usuario la respuesta deseada.

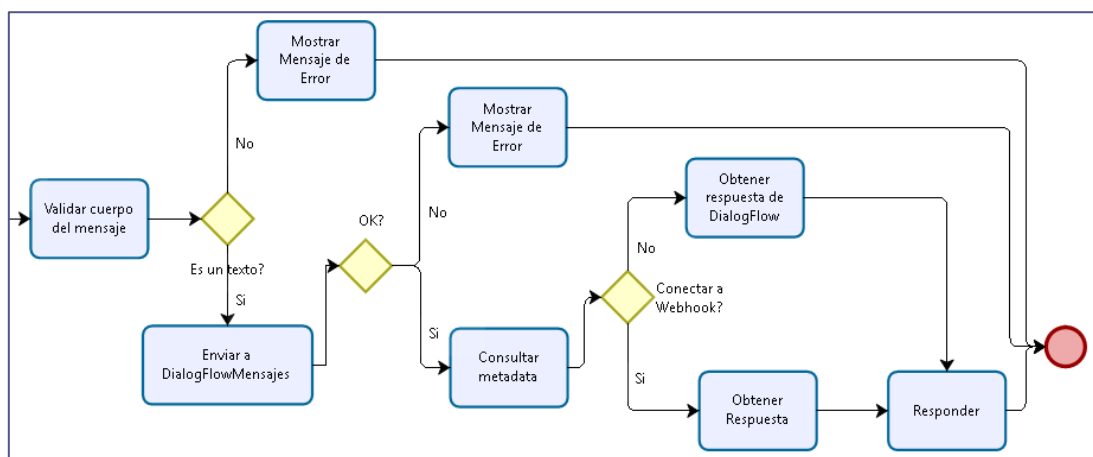


Figura 4.3. Detalle de diagrama de procesos de la integración en Telegram.
Fuente: Elaboración propia.

Módulo 2: Procesamiento del lenguaje natural

Para este módulo no es necesario escribir código, solo basta con realizar las configuraciones necesarias de la herramienta Dialogflow para que haga lo que

¹⁵ Más información en: <https://dialogflow.com/docs/reference/agent/>. Último acceso: 1 septiembre 2018.

necesitamos. Al ser una herramienta externa, funciona como una caja negra, por lo que no sabemos cómo es el diagrama de procesos. Tan solo tenemos información de cómo comunicarnos con éste mediante su API, así como de la estructura que deben tener las peticiones y las respuestas.

Para que el asistente sea bilingüe, utilizaremos un solo agente¹⁶, puesto que Dialogflow no contempla los dos idiomas. El agente lo entrenamos con preguntas escritas en valenciano y en castellano, de esta forma los algoritmos de clasificación seguirán clasificando los *Intents* correctamente. En este caso, será nuestro gestor de diálogos en el módulo 3 el que se encargue de que la respuesta al usuario final sea en el idioma correcto.

Módulo 3: Gestor de diálogo

En este módulo desarrollaremos un servicio web que en términos generales se encarga de recibir las preguntas ya clasificadas en *Intents* y, las *Entities* ya estructuradas y ordenadas. A partir de esta información realiza una consulta a la base de datos y construye la respuesta tal como la verá el usuario. Esta respuesta es mandada al gestor de Telegram para que a su vez se lo envíe al usuario correspondiente.

En la Figura 4.4 podemos observar el diagrama de procesos del servicio web del gestor de diálogos. El primer paso es esperar una petición, por eso siempre estará en el estado, escuchando peticiones. Estas peticiones llegan a través del método de peticiones HTTP POST con el formato JSON, siguiendo el formato en el que devuelve Dialogflow el texto procesado. Una vez recibido un mensaje, el siguiente paso es extraer la información contenida convirtiéndola a un formato en el que podamos trabajar con ella. En nuestro caso lo convertimos a un diccionario Python. A continuación, se lee el *Intent* de la consulta y se llama a la función que se encarga de obtener la respuesta correcta. Para ello, la función realizará los procesos necesarios para obtenerla. Una vez definida la respuesta se inserta en un JSON que tiene una cabecera que lo identifica como nuestro y se retorna la respuesta al módulo de integración de Telegram.

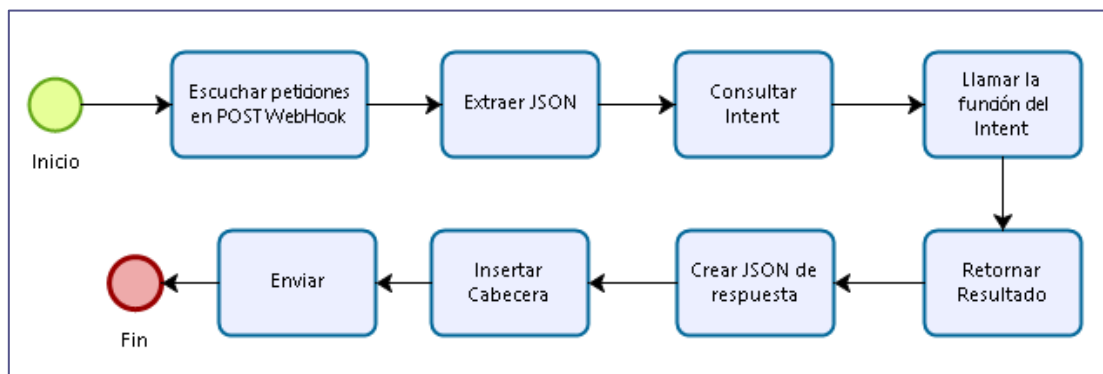


Figura 4.4. Diagrama de procesos del gestor de diálogos.
Fuente: Elaboración propia.

¹⁶ Agente es como llama Dialogflow a sus módulos de PLN.

Módulo 4. Extracción, Transformación y Carga

Este módulo está compuesto por ETL (Extracción, Transformación y Carga) que se encarga de recolectar la información y almacenarla en la base de datos que alimenta el gestor de respuestas. Se debe priorizar en el diseño de la estructura de la información en la base de datos de MongoDB, este diseño tiene que hacerse teniendo en cuenta la forma de la que va a ser consultada la información para que sea lo más eficiente posible.

Como se muestra en la Figura 4.5 la información que se necesita se encuentra en diferentes formatos: archivos en Excel, bases de datos o en páginas web. Por lo tanto, se necesita desarrollar un proceso que tome esa información, la unifique y la almacene de tal forma que resulte fácil consultar. El proceso de ETL sigue siempre los mismos pasos, comienza con la **extracción**, donde se toma de las fuentes de datos aquellos datos que son relevantes y posteriormente realiza un proceso de **transformación**, donde se da el tratamiento de los datos para que sean **cargados** en la base de datos, de tal forma que sea consultado por el módulo de gestión del diálogo.

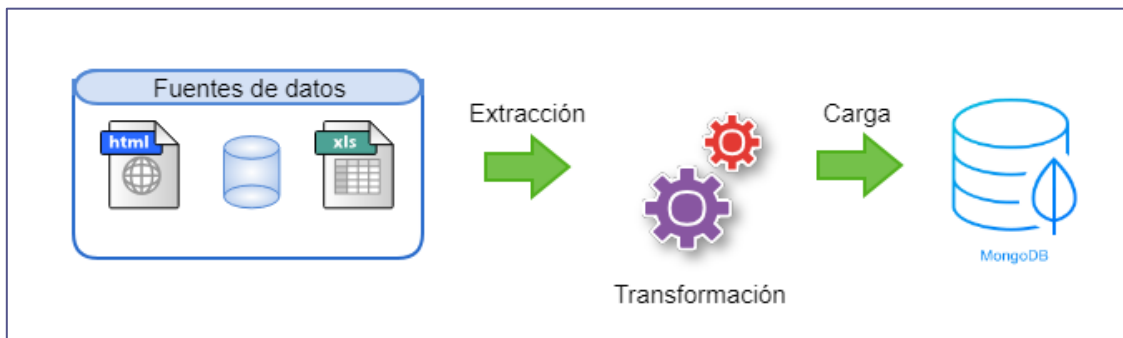


Figura 4.5. Diagrama de proceso de cada uno de los ETL.
Fuente: Elaboración propia.

Construcción

En esta etapa se realiza la construcción siguiendo lo expuesto en la fase de modelado. El módulo 1 y 3 fueron desarrollados con el lenguaje de programación Python, mientras que el módulo 3 fue desarrollado con la herramienta Dialogflow, herramienta seleccionada para crear el agente. En este módulo se rellenaron los *Intents* de saludo y de las preguntas que no tienen una clasificación definida. En lo que respecta al módulo 4, no se implementó nada durante esta iteración porque para este punto no era necesario extraer información para responder preguntas. Pasamos a ver con más detalle lo realizado en cada uno de los módulos.

Módulo 1: Integración con Telegram

Previo al desarrollo del módulo, creamos el *bot* en Telegram con el fin de reservar el nombre que queríamos utilizar, ya que el nombre debe ser único. Se realizó utilizando

“The Botfather”¹⁷, un *bot* creado por Telegram que sirve para que los usuarios creen y configuren *bots*. Después de crearlo, Botfather nos proporciona un token, que es necesario para poder escuchar los mensajes provenientes de Telegram mediante su API.

Retomando nuestro desarrollo, en la Figura 4.6 se pueden observar todos los archivos que componen el código fuente de este módulo. El archivo más importante y el que se debe ejecutar para que funcione el módulo es “telegram.py”, que utiliza la librería Telepot¹⁸ para gestionar toda la comunicación entre Telegram. Este código coordina todo el flujo del mensaje hasta que ya sabemos que es una entrada textual, por lo que entonces llamamos a una función que se encuentra en “busquedaRespuesta.py”, la cual apoyándose en las funciones que se encuentran dentro de los archivos “apiai.py”, “comunicaciónWebhook.py” y “DAO.py” realiza las comunicaciones con Dialogflow (antes llamado api.ai), el servicio web del gestor de diálogos y la base de datos respectivamente. Los archivos “botonesTeclados.py”, “textos.py” y “variables.py” contienen las variables que se utilizan en el resto de los archivos: el primero contiene la definición de todos los botones que se utilizan en Telegram; el segundo los textos de errores que contienen de forma separada; y el último contiene todas las claves necesarias para las comunicaciones.

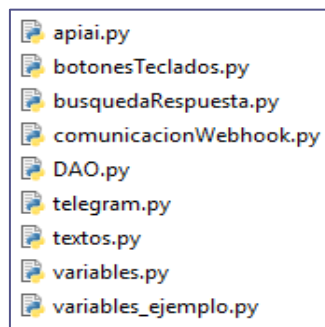


Figura 4.6. Archivos que componen el código fuente.
Fuente: Elaboración propia.

Módulo 2: Procesamiento del lenguaje natural

Al crear un agente inteligente en Dialogflow por defecto aparecen creados los dos *Intents*, “c.Saludo” que responde a entradas de saludo y “Default Fallback Intent” que responde a preguntas que no sabe clasificar dentro de ningún *Intent* de los creados. En la Figura 4.7 se puede ver los dos *Intents* creados.

¹⁷ Pasos a seguir: <https://core.telegram.org/bots#6-botfather>. Último acceso: 1 septiembre 2018.

¹⁸ Sitio web: <https://telepot.readthedocs.io/en/latest/>. Último acceso: 01 septiembre 2018.

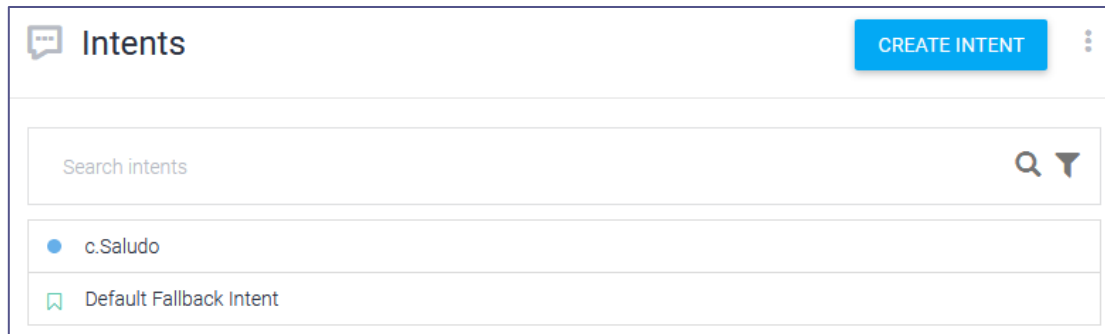


Figura 4.7. Captura del agente creado en Dialogflow donde se aprecian los Intents creados en la 1ª Iteración.
Fuente: Elaboración propia.

Módulo 3: Gestor de diálogo

En la Figura 4.8 se observan los archivos que componen el servicio web. El ejecutable de este servicio es “app.py”, que utiliza la librería Flask¹⁹ para elaborar la API REST que gestiona las comunicaciones por HTML y se encarga de la parte del framework hasta la extracción del *Intent*. Una vez extraído el *Intent*, se llama a la función de gestión de la respuesta que se encuentra dentro de “carga_respuestas.py”, que se encarga de dirigir a la función específica que genera la respuesta para ese *Intent*. Todas estas funciones las podemos encontrar en los archivos que se encuentran dentro de la carpeta “respuestas”, donde cada archivo contendrá toda lo necesario para elaborar la respuesta de un *Intent*. En esta iteración “complemento_Desconocido.py” corresponde a “Default Fallback Intent” y “complemento_Saludo.py” corresponde a “c.Saludo”. Una vez elaborada la respuesta regresara a “app.py” y este será el que se encargue de regresar el mensaje por HTTP.

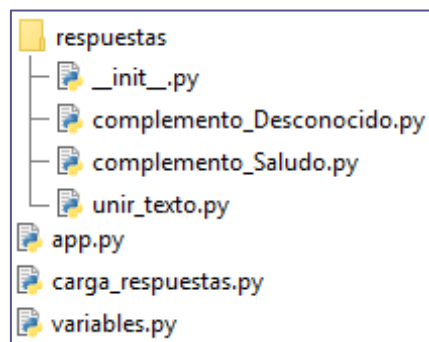


Figura 4.8. Archivos que componen el código fuente del módulo 3.
Fuente: Elaboración propia.

Evaluación y *feedback*

La evaluación por nuestra parte es positiva, se han cumplido todos los requerimientos establecidos en esta iteración: la aplicación de Telegram funciona correctamente y responde correctamente al saludo y el mensaje de error cuando no es un saludo.

¹⁹ Más información en: <http://flask.pocoo.org/>. Último acceso: 04 septiembre 2018.

Aunque en la metodología definimos que además de las evaluaciones realizadas por nosotros, haríamos pruebas que nos permitan saber la opinión de los usuarios para tener una evaluación externa. En esta primera iteración no realizamos ninguna de estas actividades para evaluar el prototipo, porque después de deliberar los resultados que podíamos obtener, consideramos que hasta que el agente no sea capaz de responder alguna pregunta a parte del saludo, no podríamos probar las funcionalidades correctamente. Por lo tanto, consideramos más adecuado dejar las pruebas para el final de la siguiente iteración.

4.2 ITERACIÓN 2

Después de diseñar y desarrollar la arquitectura, tenemos como entrada a esta iteración un asistente virtual funcional que solo sabe responder a saludos y la respuesta por defecto. En esta iteración nos dispondremos a hacer que sea capaz de responder a la primera solicitud de información.

Comunicación

En esta iteración, no realizamos un proceso de diálogo con la ciudadanía para centrarnos en la selección de la información que queremos que el asistente responda. Esto es así, porque vamos a ser pragmáticos y en esta iteración vamos a seleccionar un *dataset* que contenga la información estructurada, lo que permite que la curva de aprendizaje sea más corta y adquiriremos experiencia para más adelante centrarnos en resolver cuestiones más complicadas.

Planificación

Para esta iteración sólo añadiremos un requerimiento funcional:

Nombre	Responder a preguntas sobre cuánto se paga de impuestos por barrios.
Descripción: El sistema debe responder a preguntas sobre cuánto se paga de impuestos por barrios. Solo responde Totales, no otras operaciones.	
Precondiciones: <ul style="list-style-type: none"> • La entrada del parámetro barrio es textual • El nombre del barrio y del nombre del impuesto deberá coincidir con alguno de los sinónimos definidos 	
Flujo Normal: <ol style="list-style-type: none"> 1. El usuario envía un mensaje por alguna de las entradas admitidas. 2. El agente recibe el mensaje. 3. El agente decide que se trata de este tipo de pregunta. 	

<ol style="list-style-type: none"> 4. El agente hace una petición al servicio web en busca de la respuesta. 5. El servicio web encuentra la respuesta. 6. El servicio web manda la respuesta al agente. 7. El usuario recibe una respuesta.
<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 5. El servicio web tarda en responder. 6. El servicio web envía un mensaje de error como respuesta.
<p>Postcondiciones:</p> <ul style="list-style-type: none"> • Si el nombre del barrio se solicitará nuevamente el dato. • Si el nombre impuesto no coincide se asume que está solicitando el valor total.

Modelado

Para realizar el diseño en esta etapa, primero identificamos las fuentes de información que necesitamos para responder a la pregunta definida en los requerimientos. Acto seguido, pasamos a diseñar la base de datos de tal manera que los datos sean fácilmente accesibles por el servicio web y, finalmente realizaremos el diagrama conceptual del ETL.

La información en la que vamos a trabajar para realizar las respuestas sobre “¿cuánto se paga en impuestos en los barrios de València?”, la podemos encontrar en el subpartado “Datos por barrios” dentro del “Visor Presupuestario” en el portal de Transparencia y datos abiertos. En la Figura 4.9 podemos ver como se visualiza la información en esta página y de donde se descargan los datos.

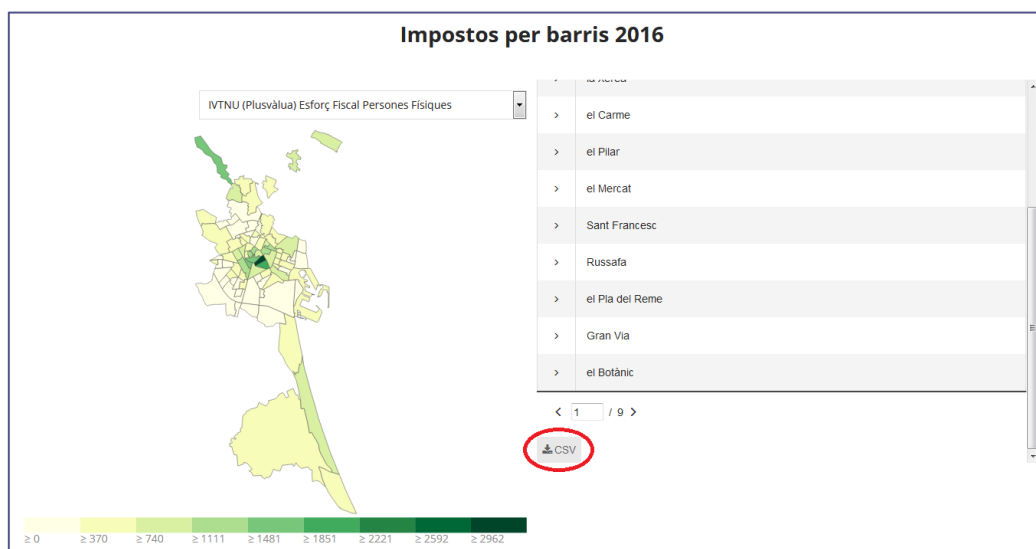


Figura 4.9. Identificación de la fuente de Información en la iteración 2.

Fuente: <http://www.valencia.es/somclars/es/#/content/58b007a73ecd372dd4529795?k=txkjaq>. Último acceso: 4 septiembre 2018.

El archivo que nos descarga es un CSV que contiene un registro por barrio. Entre los atributos encontramos el código del barrio, el nombre del barrio y el desglose de

impuestos ingresados. En la Figura 4.10 podemos ver una muestra del documento en el que se aprecian los primeros cinco registros.

1	011,la	Seu,	1284.65,87103.13,1284.65,1382.59,76574.35,2069.58,96470.77,109094.33,76.56,109.31,977167.06,556791.37,353.5336686,673.2664692
2	012,la	Xerea,	1088.7,413057.26,1088.7,12516.89,366354.29,2693.78,181496.2,144184.97,82.35,108.41,1715300.55,1041878.47,385.2894317,977.3719231
3	013,el	Carme,	575.32,129731.4,575.32,1425.62,36344.1,1397.85,205300.03,21052.43,73.3,94.83,1126325.83,549844.57,205.8720216,360.5538164
4	014,el	Pilar,	567.49,24219.72,567.49,738.93,49419.97,1453.53,142242.87,19907.56,79.73,106.46,833782.46,460287.69,233.617949,300.4488838
5	015,el	Mercat,	1188.84,108798.35,1188.84,1056.29,164065.56,2448.74,120486.03,31066.95,77.14,97.08,1158597.68,605570.95,321.6539922,441.3782434

Figura 4.10. Primeras cinco líneas del archivo CSV.

Fuente: <http://www.valencia.es/somclars/es/#/content/58b007a73ecd372dd4529795?k=txkjaq>. Último acceso: 4 septiembre 2018.

La información de este archivo la combinamos con información de tres archivos más. Los dos primeros son accesibles a través del apartado de datos abiertos del portal de València: uno contiene la división administrativa de los barrios municipales²⁰ y otro contiene la división administrativa de los distritos municipales²¹. El último archivo es de invención propia y registra todas las formas detectadas por nosotros en la que se puede llamar un barrio (Por ejemplo: el barrio del Pilar también se llama Velluters o incluso Pilar-Velluters) con estos sinónimos pretendemos uniformizar y reconocer el barrio, aunque esté escrito de distintas formas.

Una vez identificadas las fuentes de información elaboramos el diseño de la colección “impuestos_barrios”. En la Figura 4.11 podemos comprobar el resultado.

```

/* 4 */
{
  "_id" : ObjectId("5977234cbd7fa50d6ccb21f"),
  "entidad" : "barrio",
  "anio" : 2016,
  "barrio" : "EL CARME",
  "barrio_key" : "el_carme",
  "distrito" : {
    "distrito" : "CIUTAT VELLA",
    "entidad" : "distrito",
    "distrito_key" : "ciutat_vella",
    "id_distrito" : "1"
  },
  "fecha_actualizacion" : ISODate("2017-07-25T12:54:04.098Z"),
  "id_barrio" : "013",
  "sinonimos" : [
    "El Carme",
    "Carme",
    "El Carmen",
    "Carmen"
  ],
  "impuestos" : {
    "periodicidad" : "anual",
    "entidad" : "impuesto",
    "IVTNU Personas Juridicas media" : 1425.62,
    "IVTM Personas Físicas media" : 73.3,
    "IVTNU Personas Físicas" : 575.32,
    "IBI Personas Físicas" : 1126325.83,
    "IVTM Personas Juridicas media" : 94.83,
    "IVTM Personas Físicas" : 205300.03,
    "fecha_actualizacion" : ISODate("2017-07-27T12:50:15.631Z"),
    "IVTNU Personas Físicas media" : 575.32,
    "IBI Personas Juridicas media" : 360.5538164,
    "IVTM Personas Juridicas" : 21052.43,
    "IAE" : 36344.1,
    "IVTNU Personas Juridicas" : 129731.4,
    "IAE media" : 1397.85,
    "IBI Personas Juridicas" : 549844.57,
    "IBI Personas Físicas media" : 205.8720216
  }
}

```

Figura 4.11. Diseño de la base de datos 2ª iteración.

Fuente: Elaboración propia.

²⁰ Archivo en: <http://gobiernoabierto.valencia.es/es/resource/?ds=barrios&id=76d01a92-3eea-4a5c-9b5e-632d1b743421>. Último acceso: 4 septiembre 2018.

²¹ Archivo en: <http://gobiernoabierto.valencia.es/es/resource/?ds=distritos&id=23729d68-2afb-4798-931a-199218e020c6>. Último acceso: 4 septiembre 2018.

Para finalizar, solo nos queda definir el ETL, el cual se encarga de tomar la información de los cuatro archivos en formato CSV, mapearlos y cargarlos dentro de la colección definida en la Figura 4.11. El diagrama conceptual de este proceso se puede ver en la Figura 4.12.

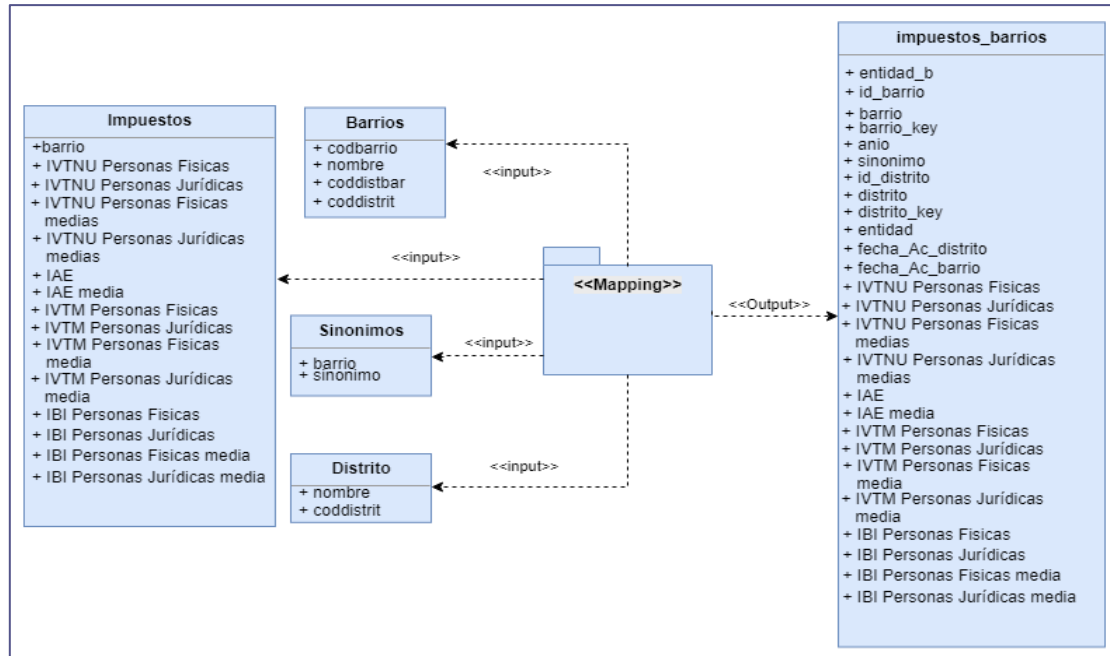


Figura 4.12. Diagrama conceptual ETL 2ª iteración.
Fuente: Elaboración propia.

Construcción

Como ya estaba definida la estructura del sistema en la iteración anterior. En esta etapa no tenemos que realizar modificaciones sobre ésta, sino que se realizan las siguientes adiciones:

- En el **módulo 2** añadimos el *Intent* “Respuesta. Pago de impuestos por barrios” y los *Entities* “Barrios” e “Impuestos”, dispuestas en las figuras Figura 4.13 y Figura 4.14.
- En el **módulo 3** añadimos las funciones necesarias en un nuevo archivo ubicado en la carpeta respuestas. Una de estas funciones se encargará de establecer la comunicación con la base de datos para consultar la información que dará respuesta a las preguntas que afecten a este *Intent*.
- En el **módulo 4** desarrollamos el ETL, con la estructura definida en la fase anterior, obteniendo como resultado una colección en la base de datos con los datos organizados como se describen en el diseño expuesto en la fase de modelado de esta iteración.

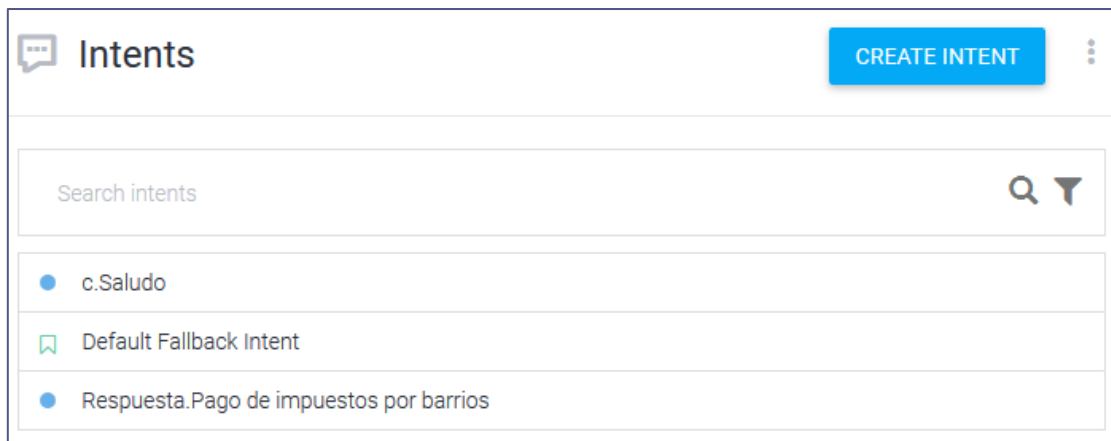


Figura 4.13. Captura del agente creado en Dialogflow donde se aprecian los Intents creados en la 2ª Iteración.
Fuente: Elaboración propia.

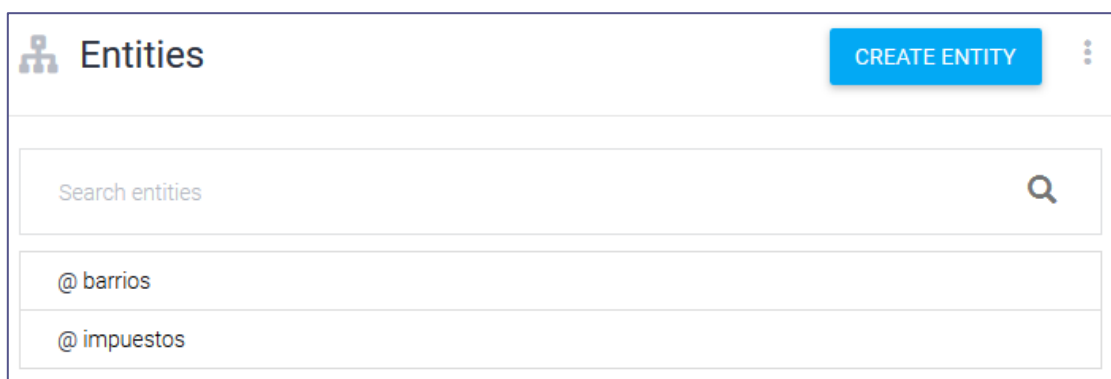


Figura 4.14. Captura del agente creado en Dialogflow donde se aprecian las Entities creadas en la 2ª Iteración.
Fuente: Elaboración propia.

Evaluación y feedback.

Después de realizar las pruebas pertinentes²² y comprobar que el nuevo *Intent* se ha insertado correctamente y que el asistente responde de forma adecuada, decidimos realizar una prueba de usabilidad para observar la reacción de usuarios reales frente al sistema.

La prueba estaba compuesta por cuatro bloques: en el primero se realizan unas preguntas previas a la prueba para saber la edad y conocimientos previos de los participantes, con el objetivo de establecer un perfil; la segunda parte gira en torno al manejo de Telegram, si sabe cómo acceder y si sabe cómo se descarga en su teléfono, en caso de no tenerlo instalado; en el tercer bloque evaluamos si una vez en Telegram son capaces de encontrar y acceder al asistente virtual; y, por último, contamos con un bloque donde la intención es ver como interactúa con el asistente, solicitando al usuario que haga unas acciones concretas. En el Apéndice B se puede ver el guion de la prueba tal cual se desarrolló.

²² Realizamos pruebas unitarias y de integración.

Con esta prueba teníamos la intención de evaluar, por un lado, las dificultades que pueden tener los usuarios para acceder al asistente, tanto el acceso a Telegram como el poder hallar el asistente una vez dentro de Telegram. Por otro lado, evaluar cómo es la interacción con el asistente, si hay problemas de usabilidad que haga que sea confuso o difícil de utilizar, o en casos más graves descubrir aspectos que hagan imposible la comunicación.

Para la ejecución de la prueba se convocó por internet a voluntarios, pero solo acudió un voluntario, por lo que los resultados no son concluyentes. No obstante, ésta nos brinda una pista de algunos problemas que podemos encontrar y que en un futuro habrá que volver a evaluar de otras formas.

En este caso, como el usuario al que le realizamos la prueba ya conocía y había utilizado Telegram, no tuvo ningún problema en su manejo. Tan solo pudimos extraer algunas reflexiones respecto a la forma de comunicarse con el asistente. El hecho curioso fue que al comunicarse con el asistente no lo hizo empleando lenguaje natural, que se emplea al comunicarse con otro humano, sino que al ser consciente que detrás del asistente había una máquina que proporcionaba las respuestas, le escribió empleando palabras clave como si escribiera en un buscador.

Esto es algo que no habíamos contemplado y que provoca que el asistente responda mal algunas cuestiones. Debemos contemplar la posibilidad de que muchos usuarios puedan comportarse de esta manera, por lo que debemos preparar al asistente para que detecte este comportamiento y le exprese el deseo al usuario de comunicarse con él como con cualquier otro contacto, o incluir este tipo de formulación en las cuestiones que entrenan los *Intent* para mejorar la detección de estas.

4.3 ITERACIÓN 3

Comunicación

Para recabar información en esta fase, simultáneamente a la convocatoria de la prueba de usabilidad, se realizó una encuesta digital para conocer qué preguntas gustaría que el asistente respondiera. Estas preguntas nos servirían, por un lado, como información de cara a nuevas funcionalidades para el asistente; y, por otro lado, para tomar ejemplos de distintas preguntas que servirán para entrenar los *Intent*.

La encuesta se realizó utilizando la herramienta Google Forms y estuvo abierta una semana. Se consiguió la participación de 44 ciudadanos, aportando un total de 220 preguntas²³.

²³ Para más información al respecto de la encuesta, puede consultar el Apéndice C.



Después del cierre de la encuesta clasificamos todas las preguntas recibidas en las siguientes categorías: Ingresos, Gastos, Presupuestos y Otros. En la Figura 4.15 se muestra el reparto de preguntas según nuestra clasificación. Puede observarse que existe un interés mayoritario en temas de Presupuestos. A la vista de los resultados se optó por priorizar este tema.

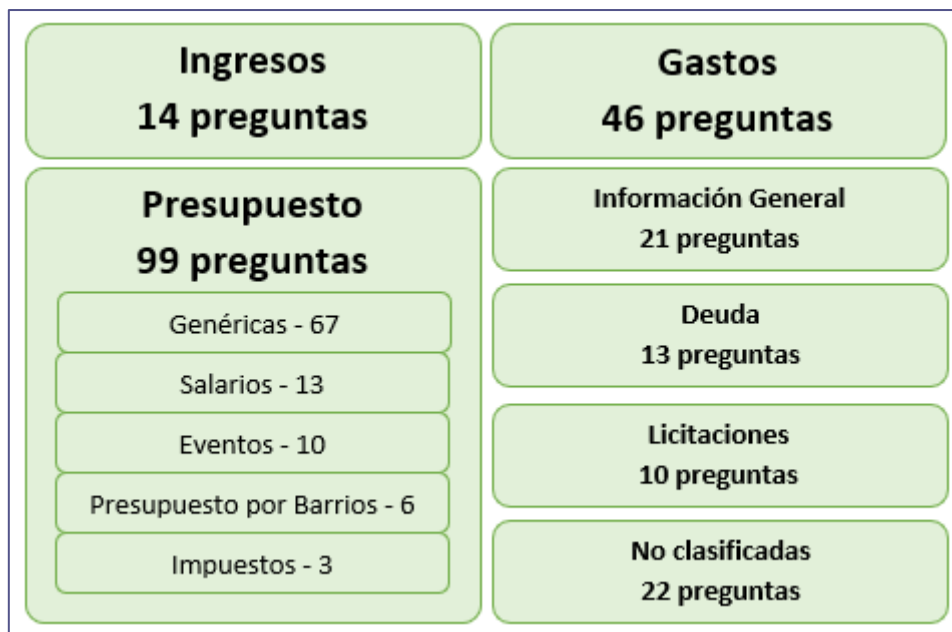


Figura 4.15. Clasificación de las preguntas obtenidas en el cuestionario de la 3ª iteración.
Fuente: Elaboración propia.

Además de la clasificación, para las preguntas que corresponden a Presupuestos analizamos si la información que solicitaban estaba disponible en los archivos publicados por el ayuntamiento y si era el caso, donde estaban publicados: en el visor, en la sección de datos abiertos o en la web del Ajuntament de València.

Después de ver este análisis, podemos observar que la mayoría de las preguntas genéricas se resuelven con datos de los presupuestos generales del ayuntamiento y otras combinando estos datos con datos de otras fuentes. En cualquiera de los casos sería necesario obtener el presupuesto en formato reutilizable. Por el momento la única forma de obtener el presupuesto en formato reutilizable es desde el visor presupuestario, pero esta información tiene un problema significativo, no está documentada, con lo cual no sabemos a qué corresponde cada campo. Por ese motivo decidimos preguntar al ayuntamiento y empezar con el siguiente subpunto dentro de los presupuestos, el salario, que también suscita gran interés.

Planificación

Para esta iteración sólo añadimos un requerimiento funcional:

Nombre	Responder a preguntas sobre el salario de los miembros de la corporación del Ajuntament de València.
Descripción: El sistema debe responder a preguntas sobre la retribución anual bruta que cobran los miembros de la corporación del Ajuntament de València cuando se pregunte por el nombre.	
Precondiciones: <ul style="list-style-type: none"> • La entrada es textual • El nombre de la persona tiene que coincidir con el del alcalde o de algún concejal, ya sea nombre o apellido. 	
Flujo Normal: <ol style="list-style-type: none"> 1. El usuario envía un mensaje por alguna de las entradas admitidas. 2. El agente recibe el mensaje. 3. El agente decide que se trata de este tipo de pregunta. 4. El agente hace una petición al servicio web en busca de la respuesta 5. El servicio web encuentra la respuesta. 6. El servicio web manda la respuesta al agente. 7. El usuario recibe una respuesta. 	
Flujo Alterno: <ol style="list-style-type: none"> 5. El servicio web tarda en responder. 6. El servicio web envía un mensaje de error como respuesta. 	
Postcondiciones: <ul style="list-style-type: none"> • Si el nombre de la persona no corresponde con ningún cargo se responde que no tenemos datos. 	

Modelado

Como en la iteración anterior, una vez claro cuál es el nuevo *intent*, en la fase de modelado seguimos la misma estructura: identificamos las fuentes de información, diseñamos la base de datos y realizamos el diagrama conceptual del ETL.

La información que necesitamos en esta iteración es: la retribución anual bruta de los concejales y el alcalde. En la Figura 4.16 podemos ver un fragmento de los datos, que se puede encontrar en la web del ayuntamiento, a través del apartado de transparencia del portal de transparencia y datos abiertos. Ya tenemos la información identificada, pero no está en ningún formato reutilizable, por lo tanto, es necesario utilizar técnicas de *web scraping* para tenerla en un formato que podamos usar.

Miembros de la Corporación. Retribución anual bruta.			
ALCALDE			
D	Joan Ribó Canut	Alcalde	82.602,94
CONCEJALES/AS			
D ^a	Sandra Gómez López	1º Teniente de Alcalde	79.298,80
D ^a .	María Oliver Sanz	2º Teniente de Alcalde	79.298,80
D ^a .	Consol Castillo Plaza	Teniente de Alcalde	74.342,66
D.	Sergi Campillo Fernández	Teniente de Alcalde	74.342,66

Figura 4.16. Fragmento de la web donde están disponibles las retribuciones brutas.

Fuente:

<http://www.valencia.es/ayuntamiento/ayuntamiento.nsf/vDocumentosTituloAux/95FFD56E2D7211EFC1257D86004964FA>. Último acceso: 1 agosto 2018

Una vez identificadas las fuentes de información, elaboramos el diseño de la colección “salarios” que se puede observar en la Figura 4.17.

```

/* 1 */
{
  "_id" : ObjectId("59771620bd7fa50d6ccbfla6"),
  "nombre" : "Joan Ribó Canut",
  "salario" : 82602.94,
  "sexo" : "Hombre",
  "cargo" : "Alcalde",
  "fecha_desde" : ISODate("2015-07-08T00:00:00.000Z"),
  "fecha_actualizacion" : ISODate("2017-07-25T11:57:52.617Z")
}
    
```

Figura 4.17. Diseño de la base de datos 3ª iteración.

Fuente: Elaboración propia.

Para finalizar, solo nos queda definir el ETL encargado de extraer la información usando web *scraping*, mapearla y cargarla dentro de la colección definida en la Figura 4.17. El diagrama conceptual de este proceso se puede ver en la Figura 4.18.

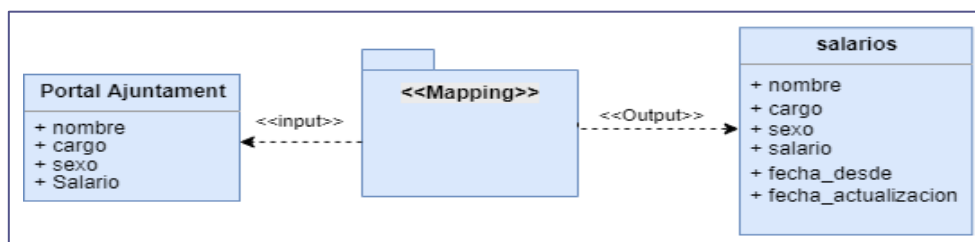


Figura 4.18. Diagrama conceptual ETL 3ª iteración.

Fuente: Elaboración propia.

Construcción

Al igual que en la iteración anterior, en el apartado de construcción realizamos las siguientes modificaciones:

- En el **módulo 2** añadimos el *Intent* “*r.salario.nombre*” como se ve en la Figura 4.19.
- En el **módulo 3** añadimos las funciones necesarias en un nuevo archivo ubicado en la carpeta respuestas encargado de consultar la información sobre los salarios en la base de datos.
- En el **módulo 4** desarrollamos el ETL, de la forma que se define en el modelado de esta iteración.

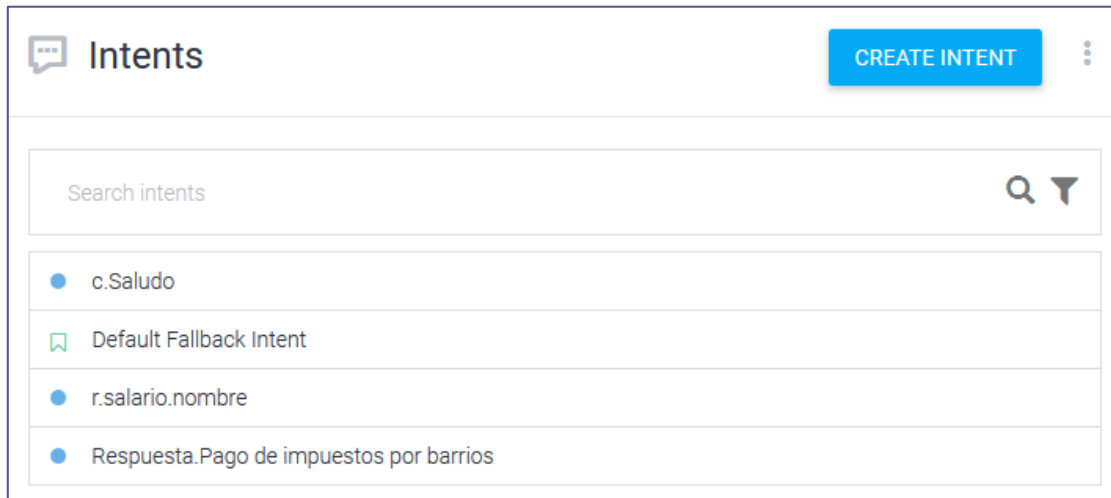


Figura 4.19. Captura del agente creado en Dialogflow donde se aprecian los Intents creados en la 3ª Iteración.
Fuente: Elaboración propia.

Evaluación y *feedback*

Realizadas las pruebas pertinentes²⁴ de nuestra parte, podemos comprobar que el *Intent* se ha insertado correctamente y el asistente responde de forma correcta tanto a las nuevas preguntas que corresponden a este *Intent*, como a las preguntas configuradas en la iteración anterior.

En otro orden de ideas, en esta iteración no realizaremos ninguna prueba adicional que implique a la ciudadanía. Al tratarse de un trabajo de fin de máster, el tiempo que disponemos para realizar todo el desarrollo tiene que ser acotado de alguna forma y tomamos la decisión de que sería en este punto. Después de utilizar las pruebas de usabilidad durante el proyecto, consideramos que este tipo de herramientas serían la forma ideal de volver a seguir con este proyecto en un futuro, si se decide darle continuidad.

²⁴ Realizamos pruebas unitarias y de integración.

4.4 RESULTADOS

Como resultado del desarrollo tenemos un asistente virtual accesible por todo el mundo que disponga de la aplicación Telegram. Para acceder solo tendrán que buscar el nombre **infoecoVLC** en el buscador. Cuando lo encuentren visualizarán una interfaz como la que aparece en la Figura 4.20 izquierda. En ésta se ve el nombre y logotipo de nuestro asistente y un pequeño texto, este primer texto sirve como primer contacto y como resumen para que los usuarios sepan qué se van a encontrar antes de pulsar iniciar.

Una vez el usuario pulse Iniciar, Telegram enviará automáticamente el comando `/start` y el asistente iniciará la conversación preguntando las preferencias de idiomas con dos botones en pantalla, que al ser pulsados mostrarán un texto confirmando la modificación. En la Figura 4.20 derecha se muestra una conversación en la que ha pasado esto, y el usuario ha pulsado el botón “Valencià”, al saludar al asistente, éste le responde en el idioma correcto y utilizando el nombre que tiene configurado en Telegram el usuario.



Figura 4.20. Capturas de inicio del asistente virtual.
Fuente: Elaboración propia.

Una vez iniciado y, siempre que el usuario no lo borre le aparecerá en su lista de contactos un contacto como el que se ve en la Figura 4.21, con el cual podrá comunicarse con nuestro asistente.

4. Desarrollo de la solución

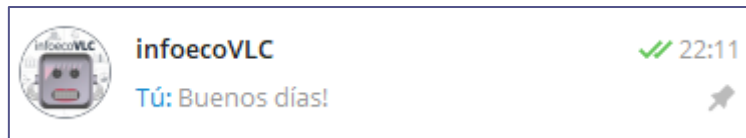


Figura 4.21. Vista del asistente en el menú de contactos de Telegram.
Fuente: Elaboración propia.

Una vez visto esto, en la Tabla 4.1 se pueden observar las conversaciones que soporta el asistente conforme está en la actualidad, los dos *Intents* que sirven para consultar la información y algunos ejemplos de los mensajes de error que se muestran en determinadas circunstancias.

Ejemplo de pregunta y respuesta sobre los impuestos que pagan los barrios.	
Ejemplo de pregunta y respuesta sobre el sueldo del alcalde.	
Mensaje que se muestra cuando alguien intenta comunicarse con el asistente virtual y este está inactivo.	
Mensaje cuando se pregunta cualquier cosa que el asistente no sepa clasificar dentro de los <i>Intents</i> contemplados.	

Tabla 4.1. Ejemplo de textos que responde el asistente.
Fuente: Elaboración propia.

5 ANÁLISIS DE RESULTADOS

Después de crear el prototipo funcional, se realiza un análisis crítico de la solución propuesta, con el objetivo de establecer los requerimientos del asistente virtual que habría que implementar. Pero antes me gustaría detenerme para destacar que el desarrollo de este Trabajo Final de Máster me ha permitido ampliar muchos conocimientos que no tenía antes de empezar su elaboración, y mejorar en el fundamento teórico-práctico tanto del estado real del *Open Data* como del desarrollo de asistentes virtuales, con todas las tecnologías implicadas. Estos conocimientos me han llevado a poder acometer un taller práctico²⁵ a fin de enseñar estos conocimientos y mostrar cómo se puede crear un asistente virtual sencillo.

La elección de Telegram para implementar la solución se ha demostrado como acertada. El API de desarrollo que ofrece Telegram es muy completa y permitió desarrollar todos los elementos que necesitaba el asistente. Además, aunque Telegram tiene muchos menos usuarios que su principal competidora, Whatsapp, cuenta con una importante comunidad de personas interesadas en la política²⁶, por lo que aun cuando sería interesante integrarlo en Whatsapp cuando este se decida a incorporar esta posibilidad, en Telegram también cumple con su función.

Un punto fuerte del desarrollo es que, antes de empezar a programar nos tomamos un buen tiempo para analizar la arquitectura con la que elaboramos la aplicación para que fuera la más óptima. Esta arquitectura tiene el punto fuerte de que permite que los módulos puedan ser intercambiados si encontramos alguna herramienta nueva o alguna forma más efectiva de hacerlos sin que afecte al resto de aplicación. Solo deberemos mantener la estructura de las comunicaciones de los módulos. Por el contrario, la parte negativa que podemos encontrar es que al unir la integración con Telegram con el gestor del *bot* nos complicaría el proceso si quisiéramos trasladar el asistente a otra plataforma.

Aún con estos puntos fuertes y débiles de la arquitectura, siempre que no cambien de forma sustancial los requisitos sobre los que partimos al hacer el modelado no debería haber grandes cambios en esta arquitectura en futuras iteraciones, y estas iteraciones se deberían centrar en incluir nuevas cuestiones que responder.

Otro punto que comentar es la decisión de utilizar Dialogflow como herramienta de procesamiento de lenguaje natural. Esta herramienta nos permitió desarrollar más rápido los prototipos y obtener una versión funcional desde la primera iteración, pero tiene un problema, que es gratuita no es *open source* y desde que fue comprada por

²⁵ Más información en el Apéndice D.

²⁶ Más información en: https://www.eldiario.es/politica/Telegram-asalta-politica-invade_0_507099699.html. Último acceso: 4 septiembre 2018.

Google la versión gratuita cada vez incorpora más limitaciones. Sería recomendable buscar alternativas *open source* o plantearse el desarrollo de este módulo por nuestra parte en un futuro para poder adaptarla más a los requerimientos propios de nuestra aplicación y para que nuestra aplicación sea íntegramente *open source* y pueda ser exportada sin depender de terceros.

En la actualidad se está desarrollando por parte de la Càtedra Govern Obert una versión 2.0 del asistente virtual intentando mejorar los puntos débiles de la actual, a fin de publicar una nueva versión que responda a más cuestiones de interés para la ciudadanía de València. Con todo, desde este trabajo definimos los siguientes requisitos que consideramos que sería interesante incluir en el futuro:

- **Insertar elementos visuales en la comunicación.** Por ahora toda la comunicación entre los usuarios y el asistente ha sido de manera textual, pero para que el usuario tenga una mejor experiencia, en las respuestas que lo amerite se deben incluir gráficas estáticas y/o gráficas interactivas.
- **Aceptar otros tipos de entrada.** La comunicación en las aplicaciones de mensajería instantánea no es sólo textual, se expresa mucha información en otros formatos como por ejemplo en emoticonos, audios entre contactos o información GPS para indicar ubicaciones. Esta información debería entenderla también el asistente.
- **Conversaciones guiadas.** Abordar temas complejos y de actualidad de forma proactiva mediante conversaciones en las que el usuario solo puede responder una de las opciones sugeridas por el propio asistente. Un ejemplo en el que se podría usar este sistema es cuando el ayuntamiento aprueba el presupuesto del próximo año. En este caso, el asistente se comunicaría con el usuario indicando que tiene esa información lista y si el usuario deseara saber más, en el caso de que la respuesta sea positiva, empezaría la conversación guiada. Con este mecanismo podremos introducir temas más complejos y de actualidad.
- **Incrementar las preguntas que pueda responder el asistente.** Si la mayoría de las preguntas que un usuario le hace el asistente, este no sabe responderle, se genera frustración en el usuario y rechazo a la aplicación. Esto puede llevar a que los usuarios abandonen la aplicación, y es posible que estas pérdidas sean imposibles de revertir.
- **Detectar usuarios que no escriban lenguaje natural.** Como se ha detectado que algunos usuarios se comunican como si el asistente virtual fuera un buscador, lo que supone un problema a la hora de procesar el lenguaje, habría que detectar este comportamiento para pedir al usuario que escriba como si escribiera a otra persona.

- **Soporte humano.** Crear un grupo de comunicación que se encargue de responder las preguntas que el asistente no sepa responder por sí mismo. Si el ayuntamiento integra el asistente dentro de sus políticas de transparencia y datos abiertos, esta opción puede impulsar el uso del asistente a la hora de responder las preguntas comunes, dejando las más atípicas en manos del grupo de comunicación. Al eliminar la opción de que el asistente virtual no entienda o no sepa responder a una pregunta, la ciudadanía se sienta más escuchada al obtener una respuesta.

6 CONCLUSIONES

A la vista de los resultados expuestos, consideramos que se ha cumplido el objetivo general de forma satisfactoria. Se ha desarrollado un prototipo plenamente funcional que nos permite acceder a la información económica y financiera del Ajuntament de València. Para que este prototipo pueda consolidar y publicarse como solución se deberá ampliar incorporando los requisitos indicados en el análisis de resultados.

El desarrollo de este Trabajo Final de Máster me ha permitido reforzar los conocimientos adquiridos durante su realización. Los conocimientos adquiridos que posteriormente fueron aplicados a lo largo del desarrollo de este proyecto fueron principalmente los relativos al estado de los datos abiertos en la administración pública vistos en las asignaturas de “Tecnologías de información para el Gobierno Abierto”, “Gestión de datos: web semántica y *open data*” y “Fuentes de datos e información”.

Aunque contaba con algunos conocimientos técnicos adquiridos previamente, gran parte del componente tecnológico empleado en este proyecto fue impulsado por las asignaturas “Explotación de datos masivos” y “Almacenamiento y recuperación de información”, la primera por los fundamentos de Python aprendidos y la segunda por los fundamentos sobre el procesamiento del lenguaje natural, que despertaron mi interés en ahondar sobre estos temas llevándome a elegir un proyecto que me permitiera desarrollarlos.

Otro aspecto relevante es que el realizar el presente trabajo en el marco de la Càtedra Govern Obert me ha permitido desenvolver el trabajo en un entorno real, manteniendo el contacto con personal técnico del Ajuntament de València cuando ha sido necesario y me ha permitido tener más poder de convocatoria cuando ha habido que implicar a la ciudadanía para recabar su opinión. Todo esto hubiera resultado imposible lograrlo por nuestros propios medios.

Con el propósito de evaluar el alcance y los logros obtenidos a partir de los objetivos específicos formulados al inicio del proyecto, podemos decir que:

- En total probamos cuatro herramientas para construir el módulo de procesamiento del lenguaje natural, donde la mejor opción fue Dialogflow. Esta herramienta sobresalió entre las demás por ser gratuita y, a la vez, permitir el uso de un servicio web para proporcionar respuestas con información almacenada en una base de datos. Esta herramienta fue muy útil para dar los primeros pasos con el asistente virtual, al permitir desarrollar de manera rápida los primeros prototipos funcionales. Pero hemos apreciado que se ha quedado un poco pequeña, porque no nos permite agregar adecuadamente algunas funcionalidades, como es el uso del valenciano.

- Como el campo de los asistentes virtuales es un campo reciente y en constante evolución, las herramientas probadas han variado mucho desde el momento en que realizamos el análisis. Aunque en su momento la herramienta más conveniente para realizar el proyecto fue Dialogflow, en este momento eso puede haber cambiado.
- Involucrar a la ciudadanía en el desarrollo ha sido una experiencia muy enriquecedora. Esto ha permitido conocer de primera mano el nivel de aceptación que tendrá la aplicación, y orientar el trabajo realizado a las necesidades reales.
- La decisión de dividir el programa en cuatro módulos diferenciados permitirá que el asistente pueda seguir creciendo y adaptándose a los nuevos tiempos con facilidad. Si surge una nueva herramienta de procesamiento de lenguaje natural o si el servicio web de gestión de respuestas se puede realizar con otras herramientas que mejoren cualquier aspecto, se podría hacer sin problemas.
- La elección de unir la gestión del *bot* con la integración del canal de Telegram no ha resultado ser una muy buena decisión. Siempre que mantengamos el asistente en Telegram no habrá ningún problema, pero en el momento que queramos migrar a otra plataforma o integrarlo con más plataformas, habrá que hacer un esfuerzo adicional y separar el código para que la integración sea correcta.
- La ciudadanía emplea de forma habitual aplicaciones de mensajería instantánea y el uso de este tipo de aplicaciones como interfaz de usuario para consultar información se ha materializado a través de la creación del asistente virtual resultante de este proyecto. Con el fin de evaluar la idoneidad de emplear Telegram con las funciones expuestas, realizamos una prueba de usabilidad al primer prototipo funcional, pero no se obtuvieron resultados concluyentes al no lograr una gran participación.
- Se ha podido comprobar, tanto en la prueba de usabilidad del asistente, como en el análisis de conversaciones recogidas por el asistente después de la realización del taller que algunos usuarios, en su interacción con el asistente virtual, modifican su forma de comunicarse. El usuario pasa a comunicarse con frases mal estructuradas o simplemente empleando palabras clave. Creemos que esto se debe a que la ciudadanía está familiarizada con máquinas de búsqueda como el buscador de Google, que normalmente utilizan este tipo de entradas. Esto en vez de ayudar, nos ocasiona problemas a la hora de interpretar los mensajes, porque les quita contexto e información.

6. Conclusiones

- A partir de la experiencia adquirida con el desarrollo de este prototipo, se han podido redactar algunos requerimientos formales que podrán guiar el desarrollo de la siguiente versión del asistente de forma más rápida y sencilla.



7 BIBLIOGRAFÍA

AIMC, 2018. *Navegantes en la Red*, Madrid: s.n.

Ajuntament de València, 2017. *Guía de transparencia para el sector público local*, València: s.n.

Ajuntament de València, 2018. *Consultes públiques amb caràcter previ a l'elaboració de projectes de normativa municipal*. [En línea]

Available at:

<https://www.valencia.es/ayuntamiento/ayuntamiento.nsf/vDocumentosTituloAux/067575EB137DE552C125828100410DE0?OpenDocument&bdOrigen=ayuntamiento%2Fauntamiento.nsf&idapoyo=967B0E31C3940650C12576700045950A&lang=2&nivel=4>

[Último acceso: 1 agosto 2018].

Bashmakov , P., 2016. *Advanced Natural Language Processing Tools for Bot Makers – LUIS, Wit.ai, Api.ai and others*. [En línea]

Available at: <https://stanfy.com/blog/advanced-natural-language-processing-tools-for-bot-makers/>

[Último acceso: 1 agosto 2018].

Berry, D., s.f. *What is Language Understanding (LUIS)?*. [En línea]

Available at: <https://docs.microsoft.com/en-us/azure/cognitive-services/luis/what-is-luis>

[Último acceso: 20 septiembre 2018].

Càtedra de Govern Obert UPV-Ajuntament de València, 2018. *Informe de Identificación y Caracterización de Iniciativas de Datos Generados por la Ciudadanía*, València:

Ajuntament de València.

CIS, 2016. *Barómetro de febrero 2016*, Madrid: s.n.

Constitución Española (1978) BOE.

Dialogflow, s.f. *Docs Dialogflow*. [En línea]

Available at: <https://dialogflow.com/docs>

[Último acceso: 20 agosto 2018].

Directiva 2003/98/CE (2003) Parlamento Europeo y del Consejo.

Directiva 2013/37/UE (2013) Parlamento Europeo y del Consejo.

Federación Española de Municipios y Provincias, 2017. *Datos Abiertos: Guía estratégica para su puesta en marcha Conjuntos de datos mínimos a publicar*. [En línea]

Available at: <http://femp.femp.es/files/3580-1617->

[fichero/Gu%C3%ADa%20Datos%20Abiertos.pdf](#)

[Último acceso: agosto 2018].

Flask, s.f. *Flask is Fun*. [En línea]

Available at: <http://flask.pocoo.org/>

[Último acceso: 20 agosto 2018].

Free Software Foundation, s.f. *GNU General Public License*. [En línea]

Available at: <https://www.gnu.org/licenses/gpl-3.0.en.html>

[Último acceso: 5 septiembre 2018].

Gil, A., 2016. La política invade Telegram. *eldiario.es*, 2 septiembre.

Hípola, P. & Vargas-Quesada, B., 1999. Agentes inteligentes: definicion y tipología. Los agentes de informacion. *El profesional de la información*, 8(4).

IAB Spain, 2017. *Estudio anual de Redes Sociales*, Madrid: s.n.

IBM Cloud, s.f. *Documentación Watson*. [En línea]

Available at: <https://console.bluemix.net/developer/watson/documentation>

[Último acceso: 20 septiembre 2018].

Iniciativa Aporta, 2018. *Iniciativa de datos abiertos del Gobierno de España*. [En línea]

Available at: <http://datos.gob.es/>

[Último acceso: 15 mayo 218].

JavaScript, 1999. *Introducing JSON*. [En línea]

Available at: <https://www.json.org/>

[Último acceso: 7 agosto 2018].

Khan, R. & Das, A., 2017. *Build Better Chatbots : A Complete Guide to Getting Started with Chatbots*. Berkeley: Apress L. P..

Ley 18/2015 (2015) BOE.

Ley 19/2013 (2013) BOE.

Ley 2/2015 (2015) Generalitat Valenciana.

Ley 37/2007 (2007) BOE.

Machiraju, S. & Modi, R., 2017. *Developing Bots with Microsoft Bots Framework : Create Intelligent Bots Using MS Bot Framework and Azure Cognitive Services*.

Berkeley: Apress L. P..

Moldovan, M. C., 2017. *Arquitecturas avanzadas de chatbots*. [En línea]

Available at: <https://labs.beeva.com/arquitecturas-avanzadas-de-chatbots-9488bcb4b218>

[Último acceso: 20 agosto 2018].

MongoDB, s.f. *Welcome to the MongoDB Docs*. [En línea]

Available at: <https://docs.mongodb.com/>

[Último acceso: 20 agosto 2018].

Nicandro Cruz-Rubio, C., 2014. *Hacia el Gobierno Abierto: Una caja de Herramientas*. s.l.:Organización de los Estados Americanos.

Norma Técnica de Interoperabilidad (2013) BOE.

Open Government Partnership, 2011. *Open Government Declaration*. [En línea]

Available at: www.opengovpartnership.org/open-government-declaration

[Último acceso: 05 mayo 2018].

Open Government Partnership, 2015. *Carta Internacional de los Datos Abiertos*. [En línea]

Available at: <https://opendatacharter.net/principles-es/>

[Último acceso: 31 julio 2018].

Open Knowledge International, 2012. *Open Data Handbook: ¿Qué son los datos abiertos?*. [En línea]

Available at: <http://opendatahandbook.org/guide/es/what-is-open-data/>

[Último acceso: 30 julio 2018].

Pintos Teigeiro, C. & Marín de la Iglesia, J. L., 2013. *Open Data. Reutilización de la información pública*. La coruña: Netbiblo SL.

Pressman, R. S., 2017. *Ingeniería del software: Un enfoque práctico*. s.l.:The McGraw-Hill Companies.

Python Software Foundation, s.f. *Python 2.7.15 documentation*. [En línea]

Available at: <https://docs.python.org/2/index.html>

[Último acceso: 20 agosto 2018].

Real Decreto 1495/2011 (2011) BOE.

Reglamento de transparencia y participación ciudadana (2015) Ajuntament de València.

Rozga, S., 2018. *Practical Bot Development: Designing and Building Bots with Node.js and Microsoft Bot Framework*. Berkley: aPress.

rss.nom, 2011. *¿Qué es RSS?*. [En línea]

Available at: <http://www.rss.nom.es/>

[Último acceso: 7 agosto 2018].

Sánchez Garreta, J. S., 2003. *Ingeniería de proyectos informáticos: actividades y procedimientos*. Castelló de la Plana: Universidad Jaume I.

Sosa, E., 1997. Procesamiento del lenguaje natural: revisión del estado actual, bases teóricas y aplicaciones (Parte I). *El profesional de la información*.

Techopedia, 2018. *Internet Bot*. [En línea]

Available at: <https://www.techopedia.com/definition/24063/internet-bot>

[Último acceso: 18 julio 2018].

Telegram, s.f. *Telegram Bot API*. [En línea]

Available at: <https://core.telegram.org/bots/api>

[Último acceso: 20 agosto 2018].

Telepot, 2018. *Introduction Telepot*. [En línea]

Available at: <https://telepot.readthedocs.io/en/latest/>

[Último acceso: 30 agosto 2018].

The Internet Society, 2005. *Common Format and MIME Type for Comma-Separated Values (CSV) Files*. [En línea]

Available at: <https://tools.ietf.org/html/rfc4180>

[Último acceso: 7 agosto 2018].

Van Lun, E., s.f. *161 Humanlike Conversational AI Synonyms*. [En línea]

Available at: <https://www.chatbots.org/synonyms/>

[Último acceso: 1 agosto 2018].

W3C, 2014. *Resource Description Framework (RDF)*. [En línea]

Available at: <https://www.w3.org/RDF/>

[Último acceso: 7 agosto 2018].

W3C, 2016. *Extensible Markup Language (XML)*. [En línea]

Available at: <https://www.w3.org/XML/>

[Último acceso: 7 agosto 2018].

wit.ai, s.f. *Getting Started with Wit.ai*. [En línea]

Available at: <https://wit.ai/docs>

[Último acceso: 20 agosto 2018].

APÉNDICE A: PRUEBA DE USABILIDAD DEL VISOR PRESUPUESTARIO

El día 2 de marzo de 2017 se reunió por parte de la Càtedra Govern Obert en el edificio de la tabacalera a un grupo de 24 personas, de las cuales 8 pertenecían a asociaciones ciudadanas y 16 a distintos departamentos del Ajuntament de València, con la intención de hacerlas una prueba de usabilidad al visor presupuestario que el Ajuntament de València iba a incorporar su portal de transparencia y datos abiertos. El objetivo de esta prueba era mejorar el producto final antes de su publicación. En la Figura A.1 podemos ver una fotografía del transcurso de este evento.



Figura A.1. Fotografía del inicio de la sesión donde se realizó la prueba de usabilidad.
Fuente: Elaboración propia.

Preguntas

Las preguntas que se formularon relevantes para este trabajo son:

1. ¿Consideraría útil disponer de un mecanismo de contacto con el Ajuntament de València?

Sí No

2. ¿Incorporaría algún sistema de búsqueda para facilitar el acceso a los contenidos?

Sí No

- En caso afirmativo, ¿cómo considera que deberían realizarse las búsquedas?
3. Si tuviera la oportunidad de que el sistema respondiera directamente a sus preguntas como lo haría una persona, ¿podría incluir la redacción de las preguntas que le formularía?
 4. ¿Valoraría positivamente poder acceder a la información presupuestaria del Ajuntament de València a través de su teléfono móvil?
 Sí No

Resultados

Los resultados obtenidos de las preguntas anteriores se insertaron en una hoja de cálculo y se interpretaron en gráficos cuando fue posible. En la Figura A.2 se pueden consultar los resultados.

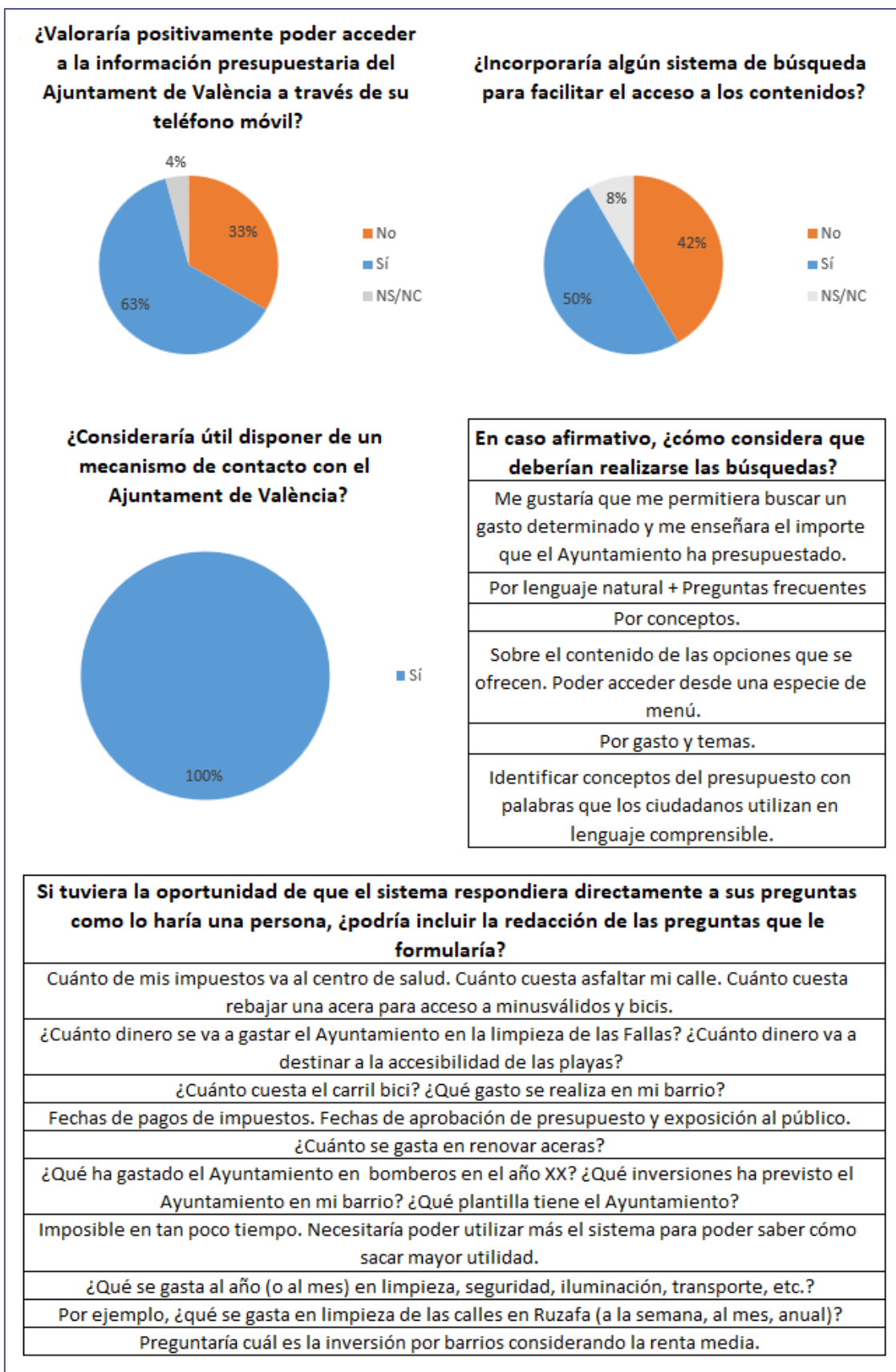


Figura A.2. Resultados prueba usabilidad visor presupuestario.
Fuente: Elaboración propia.

APÉNDICE B: PRUEBA DE USABILIDAD INFOECO VLC

Se convocó con un artículo en la web de la cátedra y difundiendo por las redes sociales a una prueba de usabilidad para el asistente virtual. La prueba se realizó el día 14 de junio de 2017 en un aula de la ETS d'Enginyeria Informàtica. En la Figura B.1 se puede observar una fotografía de la realización de la prueba.



Figura B.1. Fotografía de la realización de la prueba de usabilidad.

Fuente: <https://twitter.com/diegoalsan/status/874992330805305344>. Último acceso: 1 septiembre 2018.

La prueba se realizó por parejas usuario - evaluador, y consistía en que el evaluador iba dando indicaciones al usuario siguiendo en guion que está expuesto a continuación y se iban tomando notas de las reacciones del usuario.

Guion de la prueba de usabilidad:



BLOQUE I. Preguntas previas

1. ¿Cuántos años tiene?
 - a. Menos de 18
 - b. Entre 18 y 30
 - c. Entre 31 y 50
 - d. Más de 50

2. ¿A qué municipio pertenece?

3. ¿Utiliza Telegram como plataforma de mensajería instantánea?
 - a. Siempre
 - b. Muchas veces
 - c. Muy pocas veces
 - d. Lo he usado pero ya no lo uso
 - e. Nunca

4. En el caso de que uses Telegram ¿Qué funcionalidades utilizas?
 - a. Chat con amigos y familiares.
 - b. Comunicación con asistentes virtuales (Bots).
 - c. Seguimiento de canales de información unidireccional.
 - d. Otros: _____

5. ¿Tiene interés en temas financieros y presupuestarios de su ayuntamiento?
 - a. 1 (Nada)
 - b. 2 (Poco)
 - c. 3 (Neutro)
 - d. 4 (Bastante)
 - e. 5 (Mucho)

6. Indique en qué grado le puede resultar útil contar con una agente inteligente que le proporcione información financiera y presupuestaria de su ayuntamiento.
 - a. 1 (No es útil)
 - b. 2 (Poco útil)
 - c. 3 (Neutro)
 - d. 4 (Útil)
 - e. 5 (Muy útil)



BLOQUE II. Acceso a Telegram

1. ¿Tiene instalado la aplicación de Telegram en su dispositivo móvil?
 - a. Sí
 - b. No

En caso de no tener instalado Telegram indicar al usuario que instale Telegram, si lo tiene instalado pasar al bloque II directamente.

2. ¿Entiende que para descargar Telegram tiene que hacerlo a través del market?

3. ¿Encontró la aplicación dentro del market?

4. ¿La descargó e instaló correctamente?

5. ¿La configuro correctamente?

6. ¿Qué problemas tuvo?



BLOQUE III. Acceso a bot

Una vez instalado Telegram indicar al usuario que acceda a él y busque el bot que se llama "infoecoVLC" dentro de la aplicación.

1. ¿Encuentra dónde está el buscador?
 - a. Sí
 - b. No

2. ¿Entiende que el bot se busca como si buscara un contacto?

Cuando llegue al menú de inicio preguntar:

3. ¿Entiendes la funcionalidad del bot?
 - a. Sí
 - b. No

4. En caso negativo, preguntar: ¿Qué no entiendes?

5. ¿Entiende que se debe pulsar INICIAR para poder empezar a hablar con el bot?



BLOQUE IV. Funcionamiento bot

Para empezar este bloque tiene que estar dentro de la aplicación Telegram, haber encontrado el bot @infoecoVLCbot y haber pulsado iniciar.

En el estado actual, la agente inteligente infoecoVLC solo es capaz de responder a preguntas relacionadas con los ingresos que el ayuntamiento capta a través de los impuestos recaudados en los barrios.

Considerar contar el número de intentos fallidos que tiene el usuario al solicitar información al bot, recibiendo el mensaje por defecto que indica que el bot no logró entenderle.




1. Le pedimos al usuario que empiece a comunicarse con el bot.
 - a. ¿Saluda? ¿Cómo empieza la comunicación?

 - b. Solicitar que obtenga la cantidad de impuestos que pagan en su barrio o en otro.
 - a. ¿Identifica el nombre de su barrio? ¿Cómo ingresa la información del barrio? Si no conoce el nombre, indicarle “El Carme” como barrio por defecto para realizar la prueba.

 - b. ¿Le responde correctamente?
 - i. Sí
 - ii. No

 - c. Nro. de intentos fallidos: _____
 3. Solicitar que obtenga cuántos impuestos pagaron en su barrio en el año 2015.
 - a. ¿Le responde correctamente?
 - i. Sí
 - ii. No

 - b. Nro. de intentos fallidos: _____



4. Solicitar que obtenga la cantidad de un impuesto concreto pagado en su barrio (puede ser IBI, IAE, IVTM, IVTNU).

a. ¿Conoce los nombres de los impuestos que se pagan?

i. ¿Cuáles?

ii. ¿Alguno que no está contemplado en las fuentes de información?

b. ¿Le responde correctamente?

i. Sí

ii. No

c. Nro. de intentos fallidos: _____

5. En su comunicación con la agente inteligente el/la usuario/a:

a. ¿Utiliza frases elaboradas o escribe de manera telegráfica?

b. ¿En qué idioma formula las preguntas?

i. Castellano

ii. Valenciano

c. ¿El tono empleado por la agente inteligente le provoca alguna reacción (sonríe, mueca de desagrado, confusión, etc.)?

Página 5 de 5

Figura B.2. Formulario seguido en la prueba de usabilidad.
Fuente: Elaboración propia.

APÉNDICE C: ENCUESTA GOOGLE FORMS

La encuesta se realizó durante la semana del 08 de junio de 2017 al 15 de junio de 2017 utilizando la herramienta Google Forms y se consiguió que participaran 44 ciudadanos.

Preguntas:

Todo lo que quisiste saber sobre la gestión financiera y presupuestaria de tu ayuntamiento.

Desde la Cátedra Govern Obert estamos trabajando en el desarrollo de una agente inteligente que facilite a ciudadanos y ciudadanas el acceso a información financiera y presupuestaria de los ayuntamientos utilizando lenguaje común.

Hemos elaborado este formulario para que podáis hacernos llegar las preguntas que os gustaría que esta agente fuera capaz de responder. La agente entenderá tanto castellano como valenciano. Algunas preguntas podrían ser, por ejemplo: ¿Cuánto cobra el alcalde? ¿Cuánto dinero se invierte en mi barrio? o ¿Cuál es la deuda del ayuntamiento?

Completar la encuesta no lleva más de cinco minutos y las respuestas son totalmente anónimas.

¡Muchas gracias por colaborar!

*** Necesari**

¿Cuántos años tiene?

Menos de 18

Entre 18 y 30

Entre 31 y 50

Más de 50

¿A qué municipio pertenece?

La vostra resposta _____

¿Utiliza Telegram como plataforma de mensajería instantánea?

Siempre

Muchas veces

Muy pocas veces

Lo he usado pero ya no lo uso

Nunca

En el caso de que uses Telegram ¿Qué funcionalidades utilizas?

Chat con amigos y familiares

Comunicación con asistentes virtuales (Bots)

Seguimiento de canales de información unidireccional

Altres: _____

¿Utiliza Telegram como plataforma de mensajería instantánea?

Siempre

Muchas veces

Muy pocas veces

Lo he usado pero ya no lo uso

Nunca

En el caso de que uses Telegram ¿Qué funcionalidades utilizas?

Chat con amigos y familiares

Comunicación con asistentes virtuales (Bots)

Seguimiento de canales de información unidireccional

Altres: _____

¿Tiene interés en temas financieros y presupuestarios de su ayuntamiento? *

1 2 3 4 5

Nada Mucho

Indique en qué grado le puede resultar útil contar con una agente inteligente que le proporcione información financiera y presupuestaria de su ayuntamiento. *

1 2 3 4 5

No es útil Muy útil

Figura C.1. Formulario seguido en la prueba de usabilidad.
Fuente: Elaboración propia.

Todo lo que quisiste saber sobre la gestión financiera y presupuestaria de tu ayuntamiento.

* Necesari

Escriba las preguntas que realizaría a la agente inteligente acerca de los temas financieros y presupuestarios de su ayuntamiento. Puede indicar las preguntas en castellano o valenciano. *

Pregunta 1:

La vostra resposta _____

Pregunta 2:

La vostra resposta _____

Figura C.2. Formulario seguido en la prueba de usabilidad página 2.
Fuente: Elaboración propia.

Preguntas obtenidas

ID	PREGUNTA
1	¿De dónde obtiene ingresos el ayuntamiento?
2	Ingresos del Estado y comparativa con otras ciudades, con ratios por habitantes, porcentaje respecto al presupuesto, etc.
3	Presion fiscal
4	¿De donde procede el dinero del presupuesto municipal?
5	Relación de ingresos y gastos por barrios
6	Tienen previsto revertir los ingresos que adquieren en las terrazas y bares ocupando espacio público y ludico y produciendo molestias de suciedad y ruido a los vecinos de las zonas en algun proyecto que beneficie de algun modo a las calles afectadas?
7	Qué se recauda de la ORA y en qué se reinvierte?
8	Ingresos del Ayuntamiento
9	Conèixer les partides desl ingressos, saber com va la revalidació per conceptes
10	Els impagos,quants i en quines partides.Em referix a la falta d'ingressos per impagos.
11	Gasto i ingresos en aliances amb altres administracions públiques
12	Gasto i ingresos en aliances amb altres empreses privades en projectes socials i culturals
13	¿Cuanto dinero recibe el ayuntamiento cada año?
14	Ingresos por IAE
15	¿Cuánto dinero se destina a ayudas sociales?
16	¿Cuánto se está invirtiendo en mantenimiento y obras públicas?
17	¿Dónde se invierte más presupuesto por ciudadano?
18	¿En que porcentajes se reparte el presupuesto municipal?
19	Coste de actuaciones puntuales?
20	Presupuesto para la educación
21	En que se invierte el dinero
22	Describir mejor el presupesto para las becas.
23	¿Cuánto se invierte en cada distrito municipal?
24	Cantidad presupuestada y ejecutada en limpieza en mi municipio. Desvío respecto a la cantidad ejecutada respecto años anteriores.
25	¿Quién decide los presupuestos?
26	¿Qué distrito recibe más dinero?
27	¿Dónde se invierte menos presupuesto por ciudadano?
28	¿Cuanto dinero se destina al fomento del empleo?
29	Dinero destinado a inversiones
30	Quiero saber más acerca de los presupuestos.
31	Presupuestos destinados a jóvenes
32	¿Existe una partida destinada a la promoción de encuentros entre funcionarios y ciudadanos particulares y asociaciones?
33	¿existen partidas especiales para ayuda de personas extranjeras por razón de su nacionalidad o sólo por razones de necesidad económica? ¿cuales?
34	¿Cuáles son las partidas generales en las que se divide el presupuesto? ¿En que subpartidas se divide tal partida del presupuesto? Cantidad presupuestada y ejecutada en de tal partida o subpartida?
35	¿Que entidades municipales cuentan con presupuesto propio en mi municipio?
36	¿Cuales son los gastos de personal y su evolución en los últimos 3 años?

37	Porcentaje y cantidades asignadas a cada línea de actuación: educación, bienestar, urbanismo, juventud, etc.
38	Perceptores de subvenciones y motivo
39	¿Qué porcentaje va destinado a pagar servicios externalizados frente a los servicios públicos?
40	¿Me puedes mostrar gráficas de la evolución presupuestaria de los últimos años?
41	Dinero destinado a amortización de deuda
42	Cuál es el presupuesto de este año
43	Como está distribuido el presupuesto
44	Presupuestos y ofertas para personas desempleadas
45	Describir mejor el presupuesto para las becas.
46	¿Cuánto presupuesto recibe la planta de reciclaje?
47	¿Cuánto presupuesto se invierte en empleos temporales?
48	¿Cuánto presupuesto recibió la biblioteca el mes X del año Y?
49	¿Cuánto presupuesto se ha dado a la accesibilidad de las calles y edificios públicos?
50	¿Cuánto presupuesto recibe la policía local?
51	¿Cuánto presupuesto se destina en ayudas a la educación?
52	¿A qué se destinan (y sus porcentajes)?
53	¿Qué se hace si sobra dinero de las cuentas anuales, a qué se destina?
54	¿Cuanto dinero va destinado a temas acción social?
55	¿Que parte del presupuesto esta destinado a jóvenes emprendedores para materializar los proyectos?
56	Puedo participar en la elaboración de los presupuestos?
57	Qué porcentaje del presupuesto se destina a políticas sociales ?
58	Existen subvenciones para la sociedad civil organizada?
59	Existen mecanismos para priorizar partidas presupuestarias?
60	Porque se fracionan las partidas presupuestarias, para evitar "CONCURSO PÚBLICO"
61	Porque en los presupuestos, municipales, no existe un margen, para poder ejecutar, el sobrante no presupuestado
62	que presupuestos tienen previstos para mejorar los espacios (descampados) que hay cerrados e inutilizados (ni siquiera para aparcar) en el barrio de benicalap?
63	¿ El porqué de una obra?, presupuesto, utilidad, ¿es necesaria?
64	Quin percentatge del pressupost representa el cap 1?
65	Distribución partidas presupuestarias
66	Hay alguna partida destinada a subvencionar alguna asociacion taurina?
67	Cuánto es el volumen total de ayudas destinadas a las personas desfavorecidas entre las diferentes convocatorias que existen, y si se piensa aumentar
68	Presupostos generals desglosats
69	Insidensia dels presupost en el interes comu del poble
70	Relació amb quantitats de entitats subvencionades per l'ajuntament
71	Que parte del presupuesto se destina a mejoras en accesibilidad?
72	Presupuestan por temas, o por barrios?
73	Contemplan los habitantes del barrio para presupuestar?
74	LOS TECNICOS DEL AYUNTAMIENTO QUE HAN PRESUPUESTADO EL LA OBRAS DE LA CALLE RAMON DE PERELLOS
75	Podriem donar idees en l'elaboració del pressupot municipal?
76	¿Cómo se reparte el dinero? (Ej. 10% educación,...)

77	En qué medida podemos los ciudadanos "influir" en la elaboración de los presupuestos del Ayuntamiento?
78	Puedo participar en la elaboración de los presupuestos?
79	En qué medida podemos los ciudadanos "influir" en el control de los presupuestos del Ayuntamiento?
80	¿El ayuntamiento hace consultas de presupuestos participativos? ¿Cuándo?
81	¿Realmente se ha cumplido con los los puntos financieros y presupuestarios de los que se trataron previamente?
82	Cantidad presupuestada en mi barrio y ratio por habitante en él. Listado de barrios en orden de dicho ratio.
83	Cómo va la inversión en mi barrio
84	Presupuesto y gastos en un área concreta
85	Comparativa de inversiones por barrios
86	Qué barrios reciben menos prestaciones (transporte publico, frecuencia de limpieza, etc) a pesar de pagar sus habitantes los mismos impuestos (ibi)
87	Presupuestan por temas, o por barrios?
88	¿Cuanto dinero se gasta en fallas y en que cosas?
89	Quin es el pressupost que es dedica per a la neteja viària en dates extres (fira, setmana de bous...)
90	¿Qué importe se destina a festejos?
91	¿Cuanto dinero pagan las falleras del ayuntamiento para estar ahí?
92	Quan hem gastat en un event concret
93	¿Qué presupuesto tienen las fiestas del pueblo?
94	¿ cuanto va a costar las fiestas y actividades del pueblo?
95	¿Qué importe se destina a festejos?
96	¿Qué presupuesto tienen las fiestas del pueblo?
97	¿ cuanto va a costar las fiestas y actividades del pueblo?
98	¿Cómo se reparten los impuestos directos e indirectos?
99	Que inversión recibo a cambio
100	¿Qué porcentaje de mis impuestos se destina a la mejora de las infraestructuras de mi ciudad?
101	¿Cuánto gana al mes el alcalde?
102	Sueldo de alcalde y concejales
103	¿Cuánto gana al mes cada concejal?
104	Sueldo de un concejal detallado en sueldo bruto, dietas y demás extras
105	Dietas y alojamientos no vinculados a proyectos
106	¿Cuánto gana X funcionario?
107	¿Cuanto cobran todos y cada uno de los cargos del ayuntamiento?
108	¿ De verdad es necesario, los sueldos astronómicos, de Alcalde y Concejales?
109	Retribución alcalde y concejales
110	Está publicado sueldo y bienes, o rentas de alcalde, concejales/as, asesores,/as etc?
111	Sueldos de los representantes públicos
112	Sueldos de los funcionarios
113	Que cobran los concejales por pleno,? y ademas tienen salario?
114	Grado ejecución partidas
115	Relación entre presupuesto aprobado y presupuesto ejecutado
116	¿Qué partidas han tenido más desviación en su ejecución respecto a lo presupuestado?



117	Cantidad presupuestada y ejecutada en limpieza en mi municipio. Desvío respecto a la cantidad ejecutada respecto años anteriores.
118	¿Qué pasa con el dinero excedente?
119	Desitge conèixer l'execució de les partides pressupostàries
120	¿Realmente se ha cumplido con los los puntos financieros y presupuestarios de los que se trataron previamente?
121	Conèixer el gasto en personal fixe i temporal
122	¿Cuánto supone el sobre coste de personal (horas extra, nuevo personal...) en eventos como Semana Santa y Feria?
123	¿Cuales son los gastos de personal y su evolución en los últimos 3 años?
124	¿Cuanto dinero se gasta en fallas y en que cosas?
125	¿Cuánto cuesta el alumbrado público?
126	Cuales son los gastos actuales
127	Gasto en servicios privatizados
128	Donde van mis impuestos
129	Gastos relacionados con el transporte público
130	Relación de ingresos y gastos por barrios
131	En que se invierte el dinero
132	Cuales son los gastos actuales
133	De manera simplificada y para un ciudadano medio, ¿En que se gasta el dinero el ayuntamiento?
134	¿Números positivos o negativos? Añade la cifra.
135	¿Qué porcentaje de mis impuestos se destina a la mejora de las infraestructuras de mi ciudad?
136	Amb quines entitats financeres treballa habitualment l'ajuntament?
137	Alguna asociacion de extrema derecha recibe ayudas?
138	Con qué bancos trabaja? Con qué volumen de negocio cada uno?
139	Cuánto estamos pagando de intereses bancarios?
140	Gastos totales
141	Gastos en Educación
142	Gastos en Sanidad
143	Gastos en ayudas sociales
144	Que ens ha costat fins ara la piscina climatizada
145	A Quan ha estat els sobre costos de les obres mes importants fetes en el municipi
146	Costo de les obres fetes en el municipi.
147	Conèixer les dades amb intervals temporals per aconèixer l'evolució del gasto.
148	Sacan porcentajes de gasto/inversión por barrio?
149	CUANTO CUESTA LA LIMPIEZA DEL DISTRITO DE PATRAIX
150	CUANTO CUESTA REPARAR UNA FAROLA.
151	En que gasta el ayuntamiento los impuestos que pago?
152	Cuanto se gasta en mantenimiento del parque de mi barrio?
153	Y en limpieza? Por barrios
154	Gasto social detallat
155	Gasto i ingresos en aliances amb altres administracions públiques
156	Gasto i ingresos en aliances amb altres empreses privades en projectes socials i culturals
157	¿ cuánto dinero se invierte en el mantenimiento de zonas verdes?
158	¿ cuanto se paga por tener luz en las calles?
159	Partidas de gasto
160	Quin és el deute que té l'ajuntament a llarg, mitjà i curt termini?
161	Estructura de la deuda financiera

162	Cual es la deuda actual y el tiempo medio de pago a proveedores? ¿cual ha sido la variación de la deuda en cada uno de los últimos 3 años?
163	Deuda viva
164	¿Cuál es el nivel de endeudamiento de la ciudad?
165	¿Cuál ha sido la evolución de la deuda en los últimos 5 años?
166	¿Tiene deudas el ayuntamiento?
167	Deuda del ayuntamiento
168	¿Cuanto tiempo se necesita para que el ayuntamiento no tenga deuda?
169	Quin tipus d'interès està pagant l'ajuntament amb el deute?
170	Quin deute real té actualment l'ajuntament i quin percentatge sobre el pressupost actual suposa la seua amortització i els interesos?
171	Deuda pendiente y concepto
172	Quin deute tenim
173	Ranking de empresas contratadas por cantidad, número de contratos, por servicios
174	¿Se han modificado la realización de pliegos de contratación para que no existan sobrecostes tan escandalosos como ocurrió en la obra Metrosol (las setas)?
175	Després d'un concurs per un expedient d'inversió, en podria facilitar informació de com ha anat?
176	Quan tarda l'ajuntament en pagar als proveedors.
177	Como se hacen las licitaciones de servicios?
178	Preguntas sobre bolsas de trabajo/becas
179	Presupuestos y ofertas para personas desempleadas
180	Describir mejor el presupuesto para las becas.
181	Preguntas sobre bolsas de trabajo/becas
182	¿Qué servicios municipales cuestan más?
183	Empresas y/o fundaciones municipales
184	Porque se mantienen servicios que no son rentables
185	¿Cuales son los diferentes cargos en el ayuntamiento?
186	Porque existen, tantas designaciones "DE CONFIANZA"
187	Porque tenemos tantos INTERINOS
188	Porque no ejecutamos más OPOSICIONES, que sean necesarias, y serian funcionarios de OPOSICIÓN
189	Cuántos interinos?
190	Se contempla el ratio hombres/mujeres como funcionarios.
191	Cómo va mi factura
192	Cuantos impuestos municipales pago este año
193	Una mayor gestión, dentro de E.M.T., por las cantidad de personas, que disfrutan de un BONO, que no les corresponde
194	Com avaratar els costos en enllumenament, aigua i serveis
195	Cuántos policías tenemos con respecto a nuestro entorno?
196	¿Que instalaciones municipales cuestan más?
197	¿Cuáles son los inmuebles o espacios con los que cuenta actualmente el Ayuntamiento?
198	¿A qué se dedican los inmuebles de la pregunta anterior?
199	Carga financiera
200	lo más importante es GESTIÓN +GESTIÓN
201	¿Cuáles son las próximas actuaciones municipales?
202	¿Qué debe cada vecino hoy?
203	Porque no se gestiona, mejor, las ayudas sociales, que existen muchísimas familias, que sus ingresos, son "NEGRO", y no están contabilizados, y con grandes ingresos, reciben multitud de ayudas, perjudicando, a personas de verdad, que lo necesitan
204	Quiero que la administración, local, que es la más cercana al CIUDADANO, averigüe, datos para que bonificaciones y ayudas, sean JUSTAS, no perjudicando a terceras personas.
205	Quiero que desde nuestro AYUNTAMIENTO, exista "UN MENSAJE CIUDADANO", "NO HAY NADA GRATIS", entre todos y todas, es nuestro deber y obligación de hacer una GRANDISIMA GESTIÓN MUNICIPAL,
206	¿Porqué, no son los ciudadanos los que deciden los cambios, mediante votación?



207	Abans de començar la inversió en... S'ha fet un estudi de viabilitat?
208	¿Quien dice las prioridades para la ciudadanía?
209	Qué medidas de ahorro se están aplicando a nivel energético, consumo agua, consumo papel, mantenimiento parque informático, etc
210	Tienen contratados seguros éticos?
211	Separan por barrios la ciudad?
212	Si la ciudad es el conjunto de barrios, siempre es más importante el barrio central?
213	La medición medio ambiente, agua, aire, tierra, industria, la contemplan? La publicitan? Es tabú?
214	CUAL EL EL CANON DEL AYUNTAMIENTO POR ABRIR UNA ZANJA PARA SERVICIOS
215	Cuanta administración burocrática tenemos respecto a nuestro entorno, y media española?
216	Per què no s'inverteix en crear una xarxa de Biblioteques Públiques potents?
217	Es pensa invertir en Ciutat Vella en rehabilitació d'edificis vells per evitar la seua desaparició?
218	Pensen establir relacions amb la Generalitat i el Ministeri per continuar amb noves línies del Metro!
219	Està planificat recuperar el projecte del Metro de Ciutat Vella?
220	¿cuanto podría costar restaurar y señalizar adecuadamente las advertencias de tráfico?

APÉNDICE D: TALLER

El taller tenía como título “Taller desarrollo *chatbots* acceso información pública”, se realizó el 1 de diciembre de 2017 y la convocatoria se publicó en Eventbrite, puede consultarse en el enlace:

<https://www.eventbrite.es/e/entradas-taller-desarrollo-chatbots-acceso-informacion-publica-40844928280#>. Último acceso: 4 septiembre 2018.

La Figura D.1 se pueden una fotografía de la realización del taller.



Figura D.1. Fotografía de la realización del taller.

Fuente: <https://twitter.com/diegoalsan/status/936623828897169408>. Último acceso: 5 septiembre 2018

APÉNDICE E: ACTO DE PRESENTACIÓN DEL PROYECTO

Como el proyecto está enmarcado dentro de la Càtedra Govern Obert, el 22 de febrero de 2018 se llevó a cabo un acto en el Palau Cervelló de València con motivo de la presentación del Informe de Identificación y Caracterización de Iniciativas de Datos Generados por la Ciudadanía. Además de este informe, se presentaron todos los proyectos desarrollados en la cátedra, entre los que se encuentra el presente trabajo. En las Figura E.1 y Figura E.2 se puede observar algunas imágenes de mi presentación.



*Figura E.1. Fotografía presentación del proyecto.
Fuente: Ajuntament de València*



*Figura E.2. Fotografía presentación del proyecto.
Fuente: Ajuntament de València*