



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
Escuela de Ingeniería Informática



Elaboración de prototipos para el desarrollo de productos de Tecnología Social

TellMe

Autor: Samuel Naranjo Pérez

Tutora: Beatriz Correas Suárez

Ámbito: Tecnología Social

Grado Ingeniería Informática (Sistemas de Información)

Las Palmas, a 1 de junio de 2017

AGRADECIMIENTOS

Stefano Degano y Leticia Paz, por su colaboración en la traducción del prototipo TellMe al italiano y al francés, respectivamente.

Grimanesa Díaz y Jasmina Medina, mis leales compañeras del Curso de Adaptación al Grado, sin las cuales no hubiera llegado hasta aquí.

Beatriz Correas por su implicación y dedicación en este Trabajo Fin de Grado.

Ayoze Fernández, su ánimo y apoyo me dan las fuerzas necesarias para seguir adelante.

Mis padres, hermanos y seres queridos por su apoyo y amor incondicional.

	Página
1. Introducción	7
2. Estado actual y objetivos	8
English For Kids	8
Duolingo	9
Babbel	10
Análisis de la competencia	10
Propuesta de valor	11
3. Competencias específicas cubiertas	13
Competencias comunes del Grado	13
Competencias específicas de la mención Sistemas de Información	15
Competencias específicas del Trabajo Fin de Grado	16
4. Normativa y legislación	17
5. Aportaciones	18
Al entorno socio-económico y técnico	18
A nivel personal	18
6. Desarrollo	19
Metodología de desarrollo	19
Plan de trabajo	20
Recursos empleados	21
Estudio del sistema educativo español	22
Análisis de requisitos	24
Arquitectura del sistema	25
Seguridad	28
Diagrama de clases de TellMe Server	30
Manual operativo de usuario de TellMe Server	35
Diagrama de clases de TellMe	60
Manual operativo de usuario de TellMe	63
Proceso de comunicación entre sistemas	80

7. Conclusiones y trabajos futuros	83
8. Anexos	86
Instalación y configuración de TellMe Server en entorno desarrollo	86
Acceso al entorno candidato de TellMe Server	87
9. Fuentes de información	88

El conocimiento de una lengua extranjera permite la fácil integración de las personas en nuevos entornos, mejorando sus relaciones personales y profesionales. Además, el aprendizaje de una lengua implica la apertura a una nueva realidad, donde la persona puede participar en una sociedad globalizada y multicultural.

Las tecnologías de la información y de la comunicación (TIC) aplicadas a la educación suponen un beneficio en el proceso de enseñanza, ya que son un elemento que allanan el camino al aprendizaje y permiten conectar al alumno con los hablantes de la lengua extranjera y con su cultura.

Asimismo, los Sistemas de Información (SI) en combinación con las TIC, permiten aplicar metodologías de enseñanza abiertas, donde la persona puede asumir progresivamente la esencia de una lengua, valiéndose de innumerables recursos y facilitando un aprendizaje autónomo en contraposición con metodologías de enseñanza cerradas.

Desde un punto de vista educativo, la etapa de primaria establece los cimientos del conocimiento en el alumno. Esto le permitirá adquirir las capacidades cognitivas necesarias para asimilar, reforzar y enriquecer su aprendizaje.

Con este fin, y bajo el cumplimiento del Real Decreto 126/2014 de 28 de febrero, donde se establece el currículo básico de Educación Primaria, el área de Lenguas Extranjeras contribuye con la competencia comunicativa básica, permitiendo al alumno expresar y comprender mensajes sencillos, así como desenvolverse en situaciones cotidianas.

Por otro lado, este área coopera con el aspecto emocional del alumno; motivándolos, suscitando sentimientos de curiosidad y generando la necesidad de adquirir nuevos conocimientos de la lengua extranjera.

Dentro de este contexto, TellMe se presenta como un recurso más que ayudará a cualquier persona en el aprendizaje de la Lengua Inglesa. Para ello, este prototipo le planteará diferentes situaciones reales que le facilitarán la comprensión de textos y estructuras gramaticales.

ESTADO ACTUAL Y OBJETIVOS

Hoy en día, existen innumerables recursos disponibles en la Red que permiten el aprendizaje de un idioma, o apps desarrolladas para Tablet y Smartphone que facilitan la adquisición de dichos conocimientos. Pero no todos estos medios se ajustan a las competencias que un alumno de edad temprana debe adquirir en su educación primaria.

En concreto, para la Lengua Inglesa, nos podemos encontrar con una gran cantidad de material didáctico pero mucho de éste carece de formato u organización, lo cual dificulta la práctica del idioma, o se encuentran ancladas en metodologías de enseñanza cerradas. Por otro lado, existen otros recursos que se aproximan a los nuevos formatos de enseñanza, permitiendo entrenar las capacidades del alumno a través de ejercicios, como adivinar una palabra según una imagen, completar oraciones, etc. pero ninguno plantea situaciones reales o cotidianas donde el alumno pueda aplicar sus conocimientos.

Para poder reflejar el estado actual del aprendizaje de idiomas a través de dispositivos móviles, se ha recurrido a la plataforma de distribución de aplicaciones para Android (Google Play Store). De ella, se han elegido tres aplicaciones que han sido analizadas, lo que nos permitirá tener una mejor visión de la situación actual. Dicha elección se ha regido por el número de descargas y por la valoración de los usuarios en la plataforma.

English For Kids



Figura 1. English For Kids.

Esta aplicación educativa ha sido creada por la empresa MagikHub (Vietnam), la cual ha recibido entre 500.000 y 1.000.000 de descargas. Solo está disponible para Android y su contenido ha sido clasificado por Google Play Store como PEGI3, es decir, que ha sido considerada apta para todos los grupos de edades.

Con esta app, los usuarios podrán aprender inglés a través de vídeos que han sido agrupados por temas. Cada tema muestra una pequeña conversación subtitulada

en inglés y el objetivo consiste en que, tras visualizar el vídeo, el usuario puede reproducirlo sin audio y podrá practicar la conversación con los subtítulos.

Como valor añadido, el sistema cuenta con un chat para la comunicación entre usuarios. Está dividida en tres salas (principiante, intermedio y avanzado) y permite contactar con un usuario aleatorio o específico según sexo y edad. Además, cuenta con otros servicios como el traductor o con una sección de frases cotidianas.

Por contra, English For Kids contiene publicidad y, aunque es gratuita en todas sus opciones, tiene una opción de pago para la búsqueda específica de un usuario. Además, no se necesita crear una cuenta o autenticación de usuario, lo cual podría derivar en el uso malicioso del servicio de chat.

Duolingo



Figura 2. Duolingo.

Esta aplicación educativa ha sido desarrollada por la empresa Duolingo (Pittsburgh, Estados Unidos) y ha recibido entre 50.000.000 y 100.000.000 de descargas. Está disponible para Android, IOS y Web, y ha sido clasificado por Google Play Store como PEGI3.

Con esta app, los usuarios podrán aprender idiomas a través de diversos juegos tales como asociar una palabra a una imagen, traducir un texto o completar frases. A medida que avanza el usuario en el juego, éste conseguirá puntos.

Como valor añadido, Duolingo permite reproducir sonidos y añadir comentarios o reportar errores a cada prueba. Además, permite mostrar el significado de las palabras claves de cada prueba. Asimismo, permite conectar con otros usuarios para competir y cuenta con la sección “tienda”, donde el usuario puede utilizar sus puntos para desbloquear unidades extras o cambiar la apariencia del búho, icono representativo de la aplicación.

Por otro lado, es una aplicación gratuita y no contiene publicidad. Permite crear una cuenta de usuario y tiene un sistema de autenticación. Además, contiene una prueba de nivel y permite el aprendizaje bidireccional de los principales idiomas

(Alemán, Inglés, Español, Francés e Italiano). De esta manera, un usuario anglófono también estaría en su segmento de clientes.

Babbel



Figura 3. Babbel.

Esta aplicación educativa ha sido desarrollada por la empresa Babbel (Berlín, Alemania) y ha recibido entre 10.000.000 y 50.000.000 de descargas. Está disponible para Android y Web, y su contenido ha sido clasificada por Google Play Store como PEGI3.

El usuario puede aprender inglés asociando imágenes con palabras o completando frases con espacios en blanco. Como valor añadido, Babbel complementa su contenido explicándole al usuario los usos cotidianos de los términos dados en cada lección. Por contra, a partir de la segunda lección, el usuario debe pagar entre 9,95€ y 59,40€ por elemento.

Por otro lado, permite crear una cuenta de usuario y cuenta con un sistema de autenticación. Además, permite el aprendizaje de varios idiomas e intenta captar información del usuario a través de un cuestionario donde solicita su edad y pregunta aspectos como el objetivo del aprendizaje o cómo conoció la aplicación. De esta manera, pueden conocer mejor a su consumidor.

Análisis de la competencia

Tras examinar estas aplicaciones, podemos observar que todas contribuyen en el aprendizaje de un idioma. Además, English For Kids y Duolingo han cuidado su interfaz gráfica, haciéndolas más cómodas y amigables para niños. Se podría decir que los niños se encuentran dentro de su segmento de clientes. Por otro lado, Babbel utiliza una interfaz más formal y, al ser una aplicación de pago, buscaría clientes con un mayor poder adquisitivo.

Con respecto al modelo de enseñanza utilizado para la adquisición de conocimientos, Duolingo y Babbel utilizan metodologías de enseñanza abiertas, permitiendo al usuario desarrollar sus capacidades a través de diferentes juegos, tales como asociar una palabra a una imagen, traducir un texto o completar frases con

espacios en blanco. En cambio, English For Kids se ha quedado anclado en una metodología bastante obsoleta, "escuchar y repetir", lo cual dificulta el ejercitamiento del idioma.

Por todo ello, Duolingo es la aplicación móvil que más se aproxima a la necesidad detectada en Educación Primaria, pero sus contenidos siguen sin ajustarse a las competencias que debe adquirir un niño en su etapa educativa.

Propuesta de valor

Para poder abordar las competencias educativas de un alumno de Educación Primaria, se ha desarrollado un prototipo de apps en tecnología Android, que permita la comprensión, refuerzo y enriquecimiento de los conocimientos y competencias que el alumno irá adquiriendo durante su formación.

De esta manera, el prototipo TellMe permite acceder al usuario a multitud de contenidos didácticos que, bajo un contexto dado, podrá aplicar sus conocimientos del idioma mediante la construcción de conversaciones y, además, podrá utilizarlas en su día a día.

Inicialmente, el objetivo de la propuesta de este Trabajo Fin de Grado se dirigía a alumnos del tercer ciclo de primaria, pero tras analizar los contenidos didácticos exigidos en Educación Primaria, se ha ampliado esta propuesta a todos los ciclos de enseñanza, es decir:

- Primer Ciclo de Primaria: Primero y Segundo curso.
- Segundo Ciclo de Primaria: Tercero y Cuarto curso.
- Tercer Ciclo de Primaria: Quinto y Sexto curso.

Esta ampliación se ha considerado necesaria, ya que el aprendizaje de un idioma es progresivo y centrándonos en el último nivel, no podríamos dar cobertura a contenidos lingüísticos y gramaticales básicos.

La metodología de aprendizaje de este prototipo es muy sencilla. Cuando el usuario accede a una conversación, se le presentarán los actores que intervienen en dicho diálogo. El objetivo será ordenar las palabras de la frase que dirá cada interlocutor. Esto reforzará su destreza en la comprensión lectora. A su vez, podrá escuchar y repetir cada línea del diálogo en inglés, lo cual contribuirá a sus destrezas en comprensión auditiva y expresión oral.

Para complementar su formación, TellMe estará dotado de contenidos extras tales como aspectos gramaticales, vocabulario o expresiones relevantes que aparecerán en la línea del diálogo a modo de leyendas. Esto permitirá al usuario poder consolidar los conocimientos ejercitados en cada conversación.

Por otro lado, los diálogos han sido agrupados por temas o temáticas que, a su vez, se encuentran organizadas por niveles de dificultad. Categorías:

- Principiante: Este nivel contiene los temas y las conversaciones que cubren las competencias de los alumnos de Primer Ciclo de Primaria.
- Intermedio: Este grado alberga las temáticas y los diálogos que forjan las competencias de los alumnos de Segundo Ciclo de Primaria.
- Avanzado: Esta categoría aloja los temas y los diálogos que se adaptan a las competencias de los alumnos de Tercer Ciclo de Primaria.

Además, TellMe combina el aprendizaje del idioma con el aspecto lúdico, lo cual, permitirá motivar e incentivar al usuario. Es decir, a medida que se vayan completando conversaciones, el usuario recibirá puntos a modo de “corazones”. En el caso que descubra una leyenda, éste recibirá puntos a modo de “estrellas”. Por el contrario, si se equivoca en la ordenación de las palabras de una frase, el usuario perderá un “corazón”. Cuantas más leyendas descubra, mayor cantidad de contenidos didácticos extras tendrá disponible.

Otro aspecto que se ha tenido en cuenta es la creación de cuentas de usuario y servicios de autenticación. De esta manera, TellMe podrá almacenar la partida del jugador. Tras la validación del usuario, el prototipo habilitará dos herramientas muy importantes:

- Sugerir temas: El usuario podrá solicitar nuevas temáticas a los administradores de la aplicación. Esto permitirá tener una conexión directa con el consumidor y se podrán crear contenidos a partir de nuevas ideas.
- Notificación de erratas: Si el usuario encuentra algún contenido incorrecto, podrá comunicarlo a los administradores del sistema. Esto ayudará a corregir y mejorar los contenidos.

Estas herramientas permitirán que el alumno se exprese en inglés, practicando así la destreza en expresión escrita, o en su lengua materna. Además, para facilitar la comprensión del prototipo, TellMe será traducido a varios idiomas: Inglés, Francés, Italiano y Español. De esta manera, el usuario que sea nativo en alguno de estos idiomas, entenderá el funcionamiento de la aplicación.

Para poder dotar al prototipo de contenidos didácticos, TellMe consumirá los datos desde una aplicación Web, denominada TellMe Server, que estará alojada en un servidor. Esta herramienta extra evitará subir una nueva versión a la plataforma Google Play Store para añadir o actualizar los contenidos. TellMe Server contará con un sistema de autenticación y de permisos. Asimismo, el usuario de esta aplicación podrá crear nuevos contenidos o actualizar los ya publicados en el prototipo.

Finalmente, desde la aplicación TellMe Server, el administrador podrá atender las sugerencias y las notificaciones de erratas realizadas por los usuarios.

COMPETENCIAS ESPECÍFICAS CUBIERTAS

Competencias comunes del Grado

CII01

“Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.”

Durante el desarrollo de este trabajo, se han abordado las fases de análisis, diseño e implementación necesarias para realizar el prototipo.

CII02

“Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.”

En el anexo 1, se describen los pasos necesarios para desplegar la aplicación Web TellMe Server, así como los requisitos mínimos para su puesta en funcionamiento.

CII04

“Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.”

En el bloque Desarrollo, se especifica los requisitos de hardware necesarios para llevar a cabo este trabajo.

CII05

“Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.”

En el bloque Desarrollo, se especifica los requisitos de software necesarios para llevar a cabo este trabajo.

CII06

“Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.”

Para la arquitectura de las aplicaciones Web y móvil se ha hecho uso del patrón MVC (Modelo - Vista - Controlador) que ha facilitado su diseño, análisis e implementación.

CII07

“Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.”

Para evitar generar código poco eficiente y redundante, se han utilizado muchas estructuras de datos que han permitido minimizar y hacer entendible el código fuente, tanto en la aplicación móvil como en la aplicación Web.

CII08

“Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.”

Tras tomar la decisión de crear una aplicación Web que permita la creación y tratamiento de los contenidos didácticos a visualizar en la aplicación móvil, se analizó qué tecnologías y lenguajes de programación debían ser las adecuadas.

CII010

“Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e Implementar aplicaciones basadas en sus servicios.”

Se han tenido en cuenta las limitaciones que presentan los sistemas operativos móviles. Es por ello que se ha hecho uso de hilos para la comunicación con la aplicación Web.

CII012

“Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.”

Las estructuras de las bases de datos de ambos sistemas han sido analizadas, diseñadas y optimizadas. De esta manera, se ha evitado tener estructuras redundantes, ilegibles y poco eficientes.

CII013

“Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.”

Se han analizado, diseñado e implementado mecanismos que permiten la creación y tratamiento de los datos, así como la comunicación entre sistemas.

CII016

“Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.”

En el desarrollo de la aplicación móvil se ha tenido en cuenta el ciclo de vida de la plataforma Android. Por lo tanto, la app funciona dentro de las limitaciones de esta tecnología.

CII017

“Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.”

Ambas aplicaciones, Web y móvil, cuentan con interfaces de usuario que permiten la accesibilidad y usabilidad a todas las herramientas y servicios diseñados e implementados en este trabajo.

CII018

“Conocimiento de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional.”

Se ha tenido en cuenta la Ley Orgánica de Protección de Datos (LOPD) para garantizar y proteger el tratamiento de los datos personales de los usuarios. Asimismo, se ha tenido en cuenta el uso licencias de software.

Competencias específicas de la mención de Sistemas de Información

SI01

“Capacidad de integrar soluciones de Tecnologías de la información y las comunicaciones y procesos empresariales para satisfacer las necesidades de información de las organizaciones, permitiéndoles alcanzar sus objetivos de forma efectiva y eficiente, dándoles así ventajas competitivas.”

En este trabajo se expone una solución para cubrir una necesidad detectada en las tecnologías aplicadas a la educación; permitir que los alumnos de Educación Primaria puedan ejercitar sus conocimientos bajo un contexto dado que simule situaciones reales o familiares.

SI02

“Capacidad para determinar los requisitos de los sistemas de información y comunicación de una organización atendiendo a aspectos de seguridad y cumplimiento de la normativa y la legislación vigente.”

Dentro del bloque Desarrollo, se explican los requisitos software y hardware necesarios, así como los aspectos de seguridad aplicados para proteger la información.

SI03

“Capacidad para participar activamente en la especificación, diseño, implementación y mantenimiento de los sistemas de información y comunicación.”

Se ha especificado, diseñado e implementado una solución que permita cubrir la necesidad detectada. Asimismo, y tras analizar las competencias que un alumno de Educación Primaria debe adquirir, se ha realizado un análisis en el que se recogen los principales aspectos lingüísticos y gramaticales a los que el alumno se enfrentará durante su etapa de primaria.

SI06

“Capacidad para comprender y aplicar los principios y las técnicas de gestión de la calidad y de la innovación tecnológica en las organizaciones.”

Para el desarrollo de ambas aplicaciones, se ha hecho uso de multitud de herramientas, patrones y metodologías que satisfacen la calidad de este trabajo.

Competencias específicas del Trabajo de Fin de Grado**TFG01**

“Ejercicio original a realizar individualmente y presentar y defender ante un tribunal universitario, consistente en un proyecto en el ámbito de las tecnologías específicas de la Ingeniería en Informática de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas.”

Este trabajo ha sido analizado, diseñado e implementado por su autor.

Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal (LOPD).

Esta Ley Orgánica regula, garantiza y protege el tratamiento de los datos personales, las libertades públicas y los derechos fundamentales de las personas físicas, y especialmente de su honor, intimidad y privacidad personal y familiar.

Ley Orgánica 8/2013, de 9 de diciembre, para la mejora de la calidad educativa (LOMCE).

Esta Ley Orgánica impone nuevos parámetros en el sistema de educativo español. Lo cual modifica la Ley Orgánica 2/2006 de 3 de mayo de Educación (LOE).

Real Decreto 126/2014, de 28 de febrero, por el que se establece el currículo básico de la Educación Primaria.

Este Real Decreto establece el currículo básico de la Educación Primaria. En él, se detallan los objetivos, competencias, contenidos, criterios de evaluación y metodologías didácticas a aplicar en todos los centros escolares del territorio español.

Decreto 89/2014, de 1 de agosto, por el que se establece la ordenación y el currículo de la Educación Primaria en la Comunidad Autónoma de Canarias.

Este Decreto adapta la normativa que tiene la Comunidad Autónoma de Canarias sobre la Educación. Asimismo, establece el currículo de la Educación Primaria y regula su implantación para adaptarse con lo descrito en la LOMCE.

Licencias de Software.

La licencia de software es una especie de contrato, en donde se especifican todas las normas y cláusulas que rigen el uso de un determinado programa. Principalmente se estipulan los alcances de uso, instalación, reproducción y copia de estos productos.

Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia.

Este Real Decreto regula una serie de derechos de carácter personal y/o patrimonial que atribuyen al autor y a otros titulares, la disposición y explotación de sus obras.

Al entorno socio-económico y técnico

La revolución tecnológica ya está patente en nuestro entorno. Con sólo mirar a nuestro alrededor, podemos observar cómo la tecnología, en forma de dispositivos móviles, se mezcla con los hábitos de nuestro día a día. Como consecuencia, las empresas e instituciones públicas se han subido al carro tecnológico, creando productos para captar la atención del consumidor.

Esto ha generado que innumerables aplicaciones y servicios estén al alcance de cualquiera, facilitando la creación y tratamiento de la información. En el ámbito donde se presenta este trabajo, Tecnología Social, es fundamental que dichos servicios atiendan las demandas de una sociedad cada vez más globalizada y multicultural.

En concreto, la tecnología aplicada a la educación debe satisfacer las necesidades del alumno durante su proceso de aprendizaje. Es ahí donde el prototipo que se expone en este trabajo, permite cubrir una necesidad más y formará parte del resto de servicios que contribuyen a la adquisición de nuevos conocimientos por parte del alumno.

A nivel personal

A pesar de mis años de experiencia en el campo de la informática, no he perdido la ilusión y las ganas de seguir aprendiendo nuevas tecnologías, permitiéndome así, progresar en mi carrera profesional.

Este trabajo ha supuesto un importante reto para mí. Con él, he aprendido nuevas vertientes de la informática que me han permitido poner en marcha este prototipo. Aunque el análisis, diseño, desarrollo y mantenimiento de aplicaciones me son familiares, los aspectos relacionados con la seguridad, normativa e instalación de servicios han sido nuevos y enriquecedores.

Por último y no menos importante, mi capacidad de motivación, mi sentido de responsabilidad con el trabajo bien hecho y mi afán de autosuperación, han permitido que experimente nuevas y gratificantes sensaciones y sentimientos.

Metodología de desarrollo

Para la realización de este trabajo se ha seguido el modelo de desarrollo en espiral, el cual, consiste en dividir el ciclo de vida del software en fases. El esfuerzo del desarrollo es iterativo ya que, tras culminar una fase, da comienzo la siguiente fase. El número de iteraciones de esta metodología variará según los requisitos definidos y finalizará cuando se logre el producto mínimo viable.

Antes de comenzar una iteración, se revisarán los objetivos cumplidos en las fases anteriores. Tras este análisis, se establecerá el punto de inicio y la meta a conseguir en la nueva fase. Cada iteración generará una nueva versión del software que será más compleja que su predecesora. Cada fase está planificada con el siguiente patrón:

- **Fijar los objetivos.**
En este paso se define la meta a lograr en la fase. De esta manera, se identifican las limitaciones del proceso y sus riesgos.
- **Análisis del riesgo.**
Se analizan los riesgos que se presentan en la fase y se definen nuevas estrategias que permitan minimizar el riesgo o la eliminación de la amenaza.
- **Desarrollar, verificar y validar.**
El desarrollo es desglosado en tareas. Esto permite cubrir los objetivos fijados en la fase. Posteriormente, el equipo de desarrollo realiza las tareas planificadas y, finalmente, son probadas y verificadas.
- **Planificar.**
En este paso se revisa el proyecto y se decide si continuar con la iteración actual, donde se verificará los objetivos marcados, se revisará los requisitos del proceso y, si fuera necesario, se mejoraría la funcionalidad implementada. Tras superar este punto, se itera a una nueva fase.

Después de sucesivas iteraciones, el equipo de desarrollo habrá obtenido el producto mínimo viable que cumplirá con los requisitos definidos.

Plan de trabajo

Las tareas realizadas en este trabajo han sido desglosadas en seis fases. A continuación, se detalla la duración estimada de cada fase:

FASES	DURACIÓN (horas)	ACTIVIDADES	
Reunión con los participantes principales para hacer un análisis y una definición básica del producto.	4		
Estudio del sistema educativo español basándose en el sistema de rúbricas, los cuales contiene las competencias que el niño debe superar a lo largo del juego.	20		
Análisis de requerimientos y definición exhaustiva de la aplicación.	84	Establecer un entorno de desarrollo, junto con las tecnologías más óptimas para el desarrollo.	4
		Definir el guión del prototipo: • Historia (12) • Personajes (12) • Escenarios (12) • Recompensas obtenidas(12) • Jugabilidad inglés (32)	80
Estudio de las herramientas a utilizar para el desarrollo.	32		
Desarrollo del prototipo TellMe.	150	• Registro e inicio de sesión. • Niveles, Temas y Conversaciones. • Envío de sugerencias. • Ordenar las palabras de cada interlocutor de un diálogo.	
Desarrollo extra TellMe Server.	200	• Aplicación Web	
Desarrollo extra TellMe.	70	• Sección de Leyendas. • Envío de Erratas.	
Elaboración de la memoria.	40		
Total horas	600		

Recursos empleados

A continuación, se detallan los recursos de hardware y software utilizados para el desarrollo de este trabajo:

Hardware

- Macbook Pro con 16GB de memoria RAM, procesador Intel Core i7 a 2.3GHz y sistema operativo El Capitán.
- Teléfono móvil Huawei Mate S con 5,5” de pantalla y versión 6.0 de Android (API 23 - Marshmallow).
- Teléfono móvil Samsung S2 i9100 con 4,3” de pantalla y versión 4.1.2 de Android (API 16 - Jelly Bean).
- Tablet Alcatel OneTouch Pixi 3 con 10” de pantalla y versión 5.0.1 de Android (API 21 - Lollipop).

Software

- JDK 1.8.0.65: Paquete de herramientas para el desarrollo de aplicaciones Java.
- IntelliJ IDEA 2016.2.4: Entorno de desarrollo utilizado para la creación de la aplicaciones Web.
- Play 2.5.10: Framework de desarrollo de aplicaciones Web para Java y Scala.
- PostgreSQL: Gestor de bases de datos relacional.
- Bootstrap: Framework de CSS utilizado para la interfaz de usuario del servicio Web.
- JQuery: Framework de Javascript utilizado para la animación de la interfaz de usuario del servicio Web.
- Font Awesome: Iconografía utilizada en la aplicación Web.
- Roboto: Familia de fuentes tipográficas utilizada en el servicio Web y en el móvil.
- Android Studio 2.2.2: Entorno de desarrollo para aplicaciones Android.
- Android SDK: Paquete de herramientas para el desarrollo de apps Android.
- Genymotion 2.8.0: Emulador de dispositivos móviles.
- Realm Java 2.2.0: Gestor de bases de datos relacional en dispositivos móviles.
- Retrofit 1.9.0: Librería basada en el protocolo HTTP que permite la comunicación entre la aplicación Web y móvil.
- Adobe Photoshop CS5: Aplicación para el retoque de imágenes.
- Skech 3.8.3: Aplicación para diseño y maquetación de interfaces de usuario.
- Git 2.6.4: Software para el control de versiones.
- ArgoUML: Aplicación para la creación de diagramas estructurados.
- Open Office: Editor de textos.

Estudio del sistema educativo español

En el Real Decreto 126/2014, de 28 de febrero, se establece el currículo básico de Educación Primaria donde se regulan los objetivos, contenidos y criterios para el área de Lenguas Extranjeras. Dentro de este contexto, el sistema educativo ha definido los conocimientos y las competencias que el alumno deberá adquirir en su formación.

En consecuencia, este trabajo ha utilizado dicho marco para crear los contenidos didácticos y gramaticales que se exponen en el prototipo móvil. De esta manera, la información presentada estará adecuada a la etapa de primaria y el alumno se encontrará en un entorno conocido. Esta será la base que permitirá asimilar, reforzar y enriquecer los conocimientos que el alumno podrá utilizar en su día a día.

A continuación, se detallan los principales aspectos de la Lengua Inglesa que se imparten en la Educación Primaria.

Primer Ciclo de Primaria

En esta etapa, el alumno entra en contacto con el idioma y deberá familiarizarse con las funciones comunicativas relacionadas con el saludo, presentarse, disculparse y agradecer. Además, deberá comprender la información esencial de su entorno, al igual que identificar objetos relacionados con el aula y la escuela, reconocer juguetes e instrumentos musicales y asociar animales de granja con los productos que éstos producen.

Por otro lado, identificarán a los diferentes miembros de la familia, las partes del cuerpo, las prendas de vestir y los estados de ánimo. A su vez, podrán describir objetos mediante el color, el tamaño, etc. y podrán dar opiniones sobre los productos que les gusta y no les gusta.

Igualmente, el alumno aprenderá a reconocer lugares del exterior; la playa, el parque, la montaña, etc. y las actividades que pueden realizar en estos entornos. Asimismo, podrán asociar las partes de una casa con los objetos que contenga.

Estos conocimientos permitirán que el alumno asimile aspectos gramaticales tales como los artículos (a/an, the), los pronombres personales (I, you, etc.), y los pronombres demostrativos (this, that, etc), los pronombres posesivos (his, her, etc). Aprenderá a conjugar el presente simple y presente continuo de los principales verbos (to be, can, have got, like, want) así como las estructuras “there is” y “there are”.

Segundo Ciclo de Primaria

El alumno reforzará lo visto en la etapa anterior y aprenderá a describir a las personas y podrá hablar sobre sus hábitos y rutinas diarias. Además, profundizará en los diferentes tipos de animales; de compañía, de granja, salvajes y marinos.

Por otro lado, el alumno conocerá nuevos escenarios, como la ciudad, el camping y el circo donde podrá hablar de temas relacionados con el deporte y el turismo.

Esto permitirá que adquiera aspectos gramaticales tales como los pronombres interrogativos (what, which, etc), el uso del comparativo y superlativo, las preposiciones de lugar y tiempo (at, in, on, etc) y aplicar las estructuras “there was” y “there were”. A su vez, aprenderá a conjugar el pasado simple y el pasado continuo.

Tercer Ciclo de Primaria

En esta última fase, el alumno profundiza en el uso de los tiempos verbales del pasado (pasado simple y pasado continuo). Además, utilizará estos tiempos con verbos regulares e irregulares en oraciones positivas, negativas e interrogativas.

Por otro lado, aprenderá el futuro simple y la estructura “going to” para hablar de planes de futuro. A su vez, revisará los escenarios vistos y ampliará sus conocimientos en el uso del comparativo, el superlativo, adjetivos y adverbios de frecuencia.

Conclusiones del análisis

Tras examinar los contenidos didácticos y gramaticales que el alumno debe adquirir en su etapa de Educación Primaria, TellMe se basará en éstos para elaborar las temáticas y las conversaciones que se expondrán en el prototipo de apps.

A su vez, y con el objetivo de complementar las conversaciones creadas, se elaborarán una serie de contenidos extra, como por ejemplo, aspectos gramaticales de la lengua, mención de vocabulario relacionado con la temática o ejemplos de expresiones relevantes del idioma. Esta información estará presentada en el prototipo dentro de los apartados: unidades y leyendas.

Por consiguiente, el alumno estará familiarizado con el contexto dado y podrá ejercitar sus conocimientos. Además, podrá utilizar este material en las situaciones cotidianas que se le presenten.

Análisis de requisitos

Para cumplir con la propuesta de valor, este trabajo se ha dividido en dos partes: la aplicación Web (TellMe Server) y el prototipo de app (TellMe).

TellMe Server

Esta herramienta debe permitir la creación, consulta, edición y cancelación de los contenidos didácticos que se publicarán en la aplicación móvil. Asimismo, y dado que el prototipo cuenta con un servicio de envío de sugerencias para solicitar nuevos contenidos y con otro servicio para la detección de erratas, es necesario que el servidor almacene las peticiones realizadas por el usuario móvil. De esta manera, el usuario de TellMe Server podrá atenderlas.

Dicha aplicación deberá estar alojada en un servidor para que el prototipo TellMe consuma los contenidos didácticos publicados. Esto evitará subir una nueva versión del prototipo a la plataforma Google Play Store para añadir o actualizar los contenidos.

Por otro lado, TellMe Server debe contar con un sistema de validación de usuarios y debe regular el acceso a sus módulos a través de permisos.

TellMe

El prototipo deberá presentar los contenidos didácticos agrupados por niveles de dificultad (principiante, intermedio y avanzado). Cada nivel, deberá agrupar las conversaciones en temas o temáticas.

El usuario deberá acceder a una conversación y ordenar las palabras de la frase de cada interlocutor. El sistema deberá validarla, así como sumar o restar los puntos que el usuario vaya consiguiendo. Asimismo, si la conversación tiene alguna leyenda oculta, ésta deberá mostrarse y añadirse a la cuenta del jugador. Si el usuario pierde todos los puntos, el sistema deberá permitir reiniciar la partida.

Por otro lado, el prototipo debe contar con un sistema de validación de usuarios y deberá almacenar la partida del jugador; conversaciones realizadas, leyendas descubiertas y puntos conseguidos.

Arquitectura del sistema

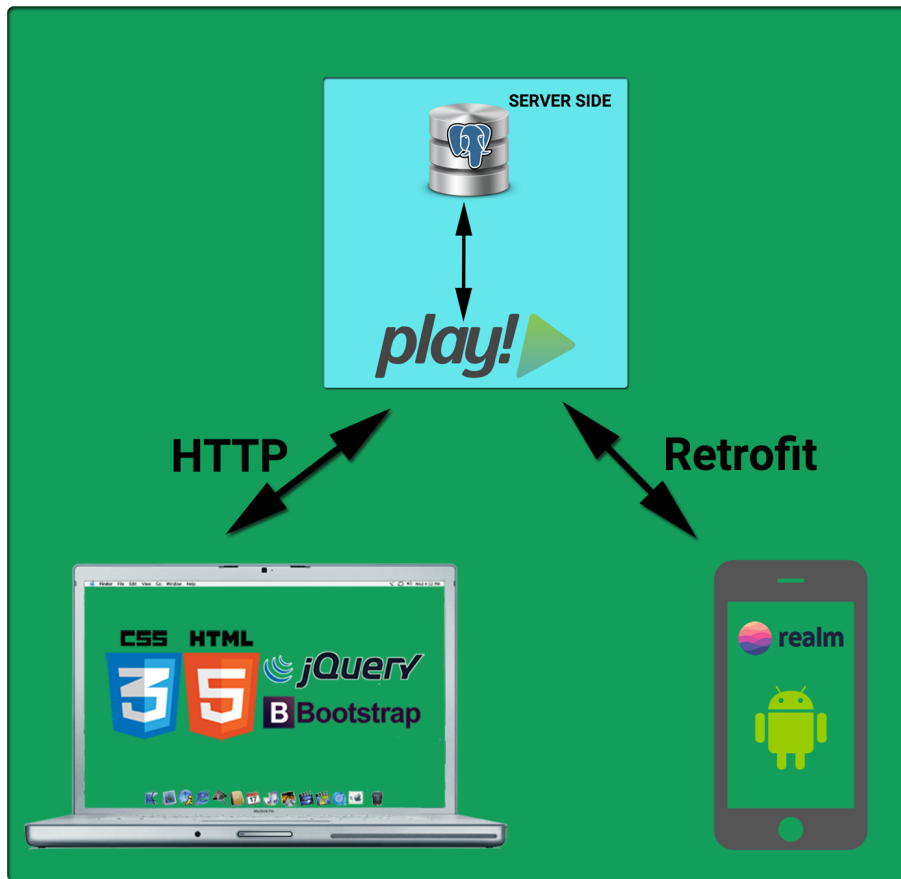


Figura 4. Arquitectura.

Para poder dar cobertura a las necesidades descritas en el apartado anterior, TellMe Server debe ser una aplicación Web que permita al usuario interactuar con la información. Para ello, se ha elegido a “Play”, un framework de desarrollo web para Java y Scala basado en el patrón MVC (Modelo, Vista, Controlador).

Además, Play permite conexiones REST; un estilo de arquitectura para desarrollar servicios web que utiliza el estándar HTTP y JSON para la transmisión de datos. REST sigue la arquitectura cliente-servidor, en la cual, un cliente realiza peticiones al servidor y espera a que le de respuesta. De esta manera, la aplicación móvil podrá comunicarse con TellMe Server.

Por otro lado, la información debe ser almacenada en una base de datos. Por tanto, el servidor debe contar con un sistema de gestión de base de datos. Para ello, se ha utilizado “PostgreSQL”; un potente gestor de bases de datos relacional.

Para el diseño de la interfaz gráfica, se ha optado por Bootstrap; un framework de CSS que, en combinación con HTML y JQuery, permite un desarrollo web más rápido y fácil.

Además es “responsive design”, lo que permite adaptar el sitio web al tamaño de la pantalla del dispositivo en el que se esté visualizando. Junto con Bootstrap, la

interfaz ha sido dotada con “Font Awesome”; una fuente de iconos que permitirá al usuario identificar fácilmente la usabilidad de los componentes de la interfaz.

Por último, TellMe Server cuenta con su propia tipografía. Para ello, se hace uso de la tipografía “Roboto”; una familia de fuentes que sobrescribirá la fuente que usan los navegadores por defecto. De esta manera, la aplicación Web se verá igual en todos los navegadores.

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
2.3 Gingerbread	10	
4.0 Ice Cream Sandwich	15	97,4%
4.1 Jelly Bean	16	95,2%
4.2 Jelly Bean	17	87,4%
4.3 Jelly Bean	18	76,9%
4.4 KitKat	19	73,9%
5.0 Lollipop	21	40,5%
5.1 Lollipop	22	24,1%
6.0 Marshmallow	23	4,7%

Figura 5. Versiones de Android.

El prototipo de app abarca desde la versión 16 hasta la versión 23. De esta manera, TellMe funcionará para el 95,2% de los terminales móviles del mercado. Asimismo, se ha hecho uso de la guía “Material Design” de Google para el diseño de las pantallas. Dicho manual establece las recomendaciones para diseñar los componentes gráficos, así como el uso de colores, dimensiones, márgenes, posicionamiento de los elementos, etc.

Por lo general, Android utiliza SQLite para almacenar la información pero este prototipo lo ha descartado y ha utilizado la librería “Realm IO”; un potente gestor de bases de datos relacional que destaca sobre sus competidores por la velocidad de acceso a los datos.

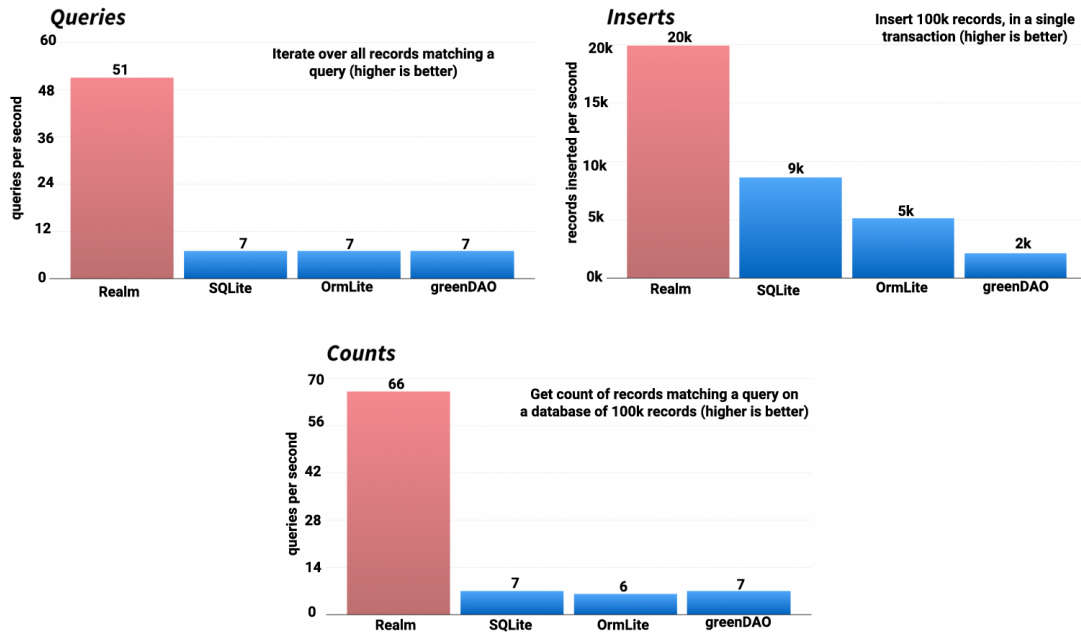


Figura 6. Comparación en tiempos de ejecución entre diferentes tecnologías.

Por último, para la comunicación entre ambos sistemas, se ha utilizado la librería “Retrofit”. Es un cliente REST que permite hacer peticiones GET, POST, PUT o DELETE. De esta manera, el prototipo podrá conectar con la aplicación web y consumir los datos necesarios.

Además, esta librería permite gestionar diferentes tipos de parámetros, la descarga de objetos JSON y XML y parsear la respuesta a un objeto Java. Para reforzar esta librería, se ha utilizado la librería “Gson”, que permite dicho parseo de objetos y, por ejemplo, si un JSON tuviera más atributos que su homólogo en Java, el parseo se realizaría correctamente y sólo rellenaría los atributos que tuviera declarado el objeto Java.

Seguridad

A continuación, se describen los mecanismos de seguridad utilizados:

TellMe Server

Protección contra Cross Site Request Forgery (CSRF): Este ataque fuerza al navegador web de su víctima a enviar una petición a una aplicación web vulnerable, lo cual realizaría la acción maliciosa a través de su víctima. Para evitar esto, Play incluye un token a sus formularios. De esta manera, cuando se realiza la acción, Play valida si el token es suyo y permite realizar la acción.

```
▼<form action="/login" method="POST">
  <input type="hidden" name="csrfToken" value=
    "7b70000bd4f70abfda5493c13328ecf6d40b81a7-1494258846846-aa7fdedb205664ef7f36cebf">
  <div class="form-group">...</div>
  <div class="form-group">...</div>
  <button type="submit" class="btn btn-primary showWaiting">INICIAR SESIÓN</button>
</form>
```

Figura 7. Protección contra CSRF.

Doble validación: Play permite configurar el tipo de peticiones HTTP que no necesitan ser verificadas. En este prototipo se ha habilitado el tipo GET para que el usuario pueda acceder al index de la página de inicio de la aplicación. El resto de peticiones (POST, PUT, etc) y los demás métodos GET pasarán por la siguiente validación.

```
# Method lists
method {
  # If non empty, then requests will be checked if the method is not in this list.
  whitelist = ["GET"]
  # The black list is only used if the white list is empty.
  # Only check methods in this list.
  blacklist = []
}
```

Figura 8. Control de los tipos de peticiones.

Play utiliza esta anotación `@Security.Authenticated` para comprobar si el usuario está registrado en la sesión del navegador. En caso contrario, retornará el error 401.

```
/**
 * Metodo utilizado para obtener usuario autenticado
 * @return
 */
@Security.Authenticated(Secured.class)
public static String currentUser() {
  try {
    if(session(CommonController.SessionKeyEnum.USER.getConstant())!=null) {
      LoginDTO loginDTO = Json.fromJson(Json.parse(session(CommonController.SessionKeyEnum.USER.getConstant()))), LoginDTO.class);
      return loginDTO != null ? loginDTO.currentUser() : null;
    }
  } catch (Throwable t) {
    Logger.error(null, t);
  }
  return null;
}
```

Figura 9. Anotación para determinar si un usuario tiene acceso al método.

TellMe

Encriptación de la base de datos: La librería “Realm IO” permite encriptar la base de datos donde se almacenarán los datos del prototipo así como los datos del usuario; cuenta de email, nombre de usuario y contraseña. Debido a que se puede acceder desde el exterior, se ha generado una clave que encripta el contenido.


```

Realm.init(context);
configuration = new RealmConfiguration.Builder()
    .name("tellme.realm")
    .encryptionKey(context.getString(R.string.encrypt).getBytes())
    .schemaVersion(1)
    .migration(new TellMeMigration())
    .build();
Realm.setDefaultConfiguration(configuration);

```

Figura 10. Encriptación de la BBDD del prototipo.

Comunicación con TellMe Server: A través de la librería “Retrofit”, el prototipo se podrá conectar con la aplicación Web para consumir los contenidos didácticos, así como reportar sugerencias y erratas. Para ello, se deben declarar los siguientes métodos:

```

public interface Service {

    @GET("/external/publications/{id}")
    void getNewPublications(@Path("id") Long id, Callback<NewPublication> cb);

    @GET("/external/updates/{id}")
    void getUpdates(@Path("id") Long id, Callback<UpdatePublication> cb);

    @POST("/external/suggestion")
    void postSuggestion(@Body SendSuggestion sendSuggestion, Callback<SendSuggestion> cb);

    @POST("/external/mistake")
    void postMistake(@Body SendMistake sendMistake, Callback<SendMistake> cb);
}

```

Figura 11. Declaración de métodos.

Para consumir los datos, se debe construir una instancia de RestAdapter, a la cual se le pasa el host y el nombre de usuario y contraseña para la autenticación. En un periodo máximo de 90 segundos se recibirá la respuesta JSON que es parseada a un objeto Java. Lo ideal sería que el servidor tuviera habilitado el protocolo SSL ya que, al establecer la conexión, ésta se cifra. A partir de ese momento, el sistema podrá utilizar los datos obtenidos. Si la conexión tarda más de 90 segundos, la instancia RestAdapter aborta la petición.

```

OkHttpClient okHttpClient = new OkHttpClient();
okHttpClient.setReadTimeout(90, TimeUnit.SECONDS);
okHttpClient.setConnectTimeout(90, TimeUnit.SECONDS);
RestAdapter.Builder builder = new RestAdapter.Builder().setClient(new OkClient(okHttpClient));
builder.setEndpoint(getHostConnection());
builder.setRequestInterceptor(new RequestInterceptor() {
    @Override
    public void intercept(RequestFacade request) {
        request.addHeader("Authorization", getUserConnection());
        request.addHeader("Accept", "application/json");
    }
});
RestAdapter adapter = builder.build();
Service service = adapter.create(Service.class);
if (service != null) {
    service.getNewPublications(id, new Callback<NewPublication>() {
        @Override
        public void success(NewPublication newPublication, Response response) {
            if (newPublication != null) {
                //Do something
            }
        }
        @Override
        public void failure(RetrofitError error) {
            error.printStackTrace();
        }
    });
}
}

```

Diagrama de anotaciones en la Figura 12:

- Tiempo máximo de conexión:** Señala a los valores 90 en `setReadTimeout` y `setConnectTimeout`.
- Conexión al servidor:** Señala a `builder.setEndpoint(getHostConnection());`.
- Usuario y contraseña de acceso:** Señala a `request.addHeader("Authorization", getUserConnection());`.
- Método:** Señala a `service.getNewPublications(id, new Callback<NewPublication>() {`.
- Respuesta OK:** Señala a la rama `success` del callback.
- Respuesta no OK:** Señala a la rama `failure` del callback.

Figura 12. Ejemplo de conexión a la aplicación Web.

Diagrama de clases de TellMe Server

A continuación, se detallan las entidades utilizadas en la aplicación TellMe Server.

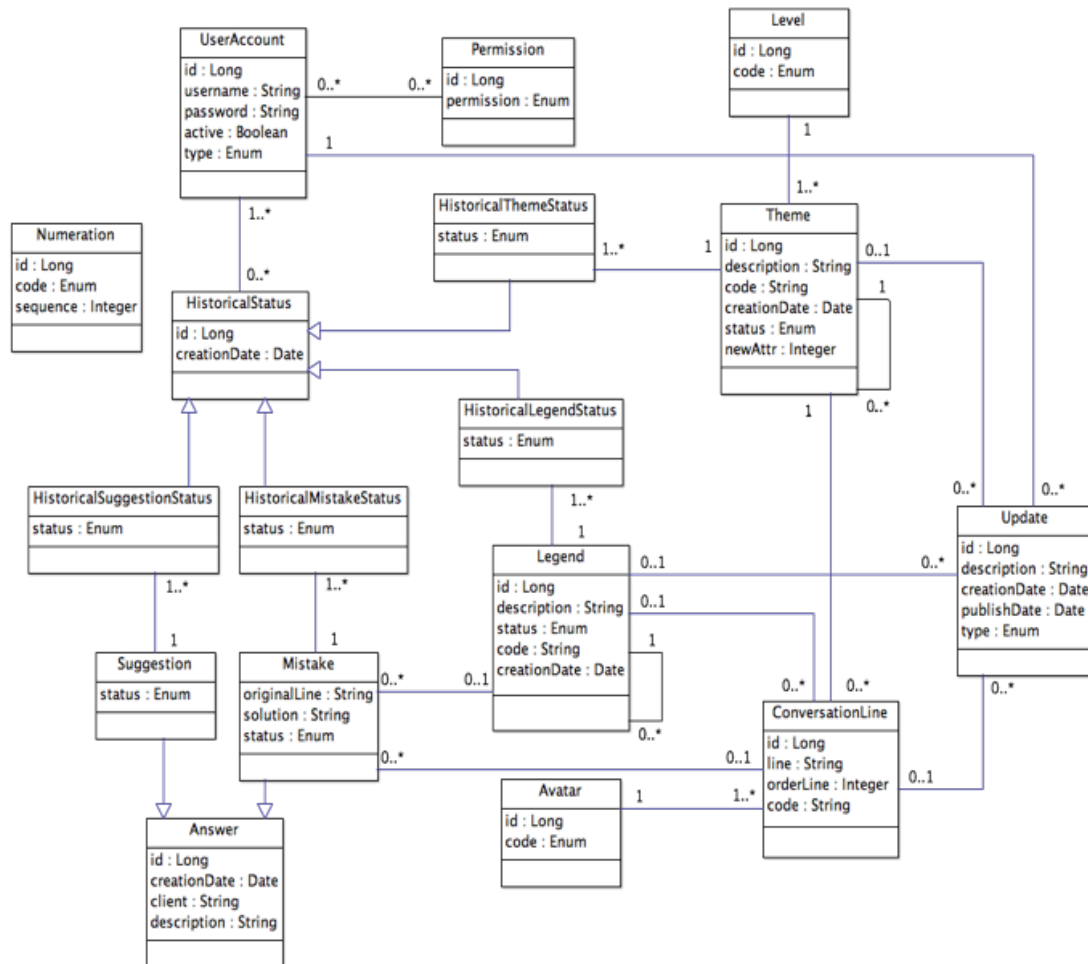


Figura 13. Diagrama de clases de TellMe Server.

Clase: UserAccount	
Descripción: Clase utilizada para almacenar las cuentas de usuario.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
username String	Identificador del usuario. Es único y obligatorio.
password String	Clave de autenticación del usuario. Está cifrado y es obligatorio.
active Boolean	Determina si el usuario tiene acceso al sistema. Es obligatorio.
type Enum	Campo obligatorio y diferencia el tipo de acceso del usuario.
Los tipos de acceso definidos son: • WEB: Acceso a través de la interfaz gráfica de la aplicación. • COM: Acceso a través de los servicios de comunicación disponibles.	

Clase: Permission	
Descripción: Clase utilizada para almacenar los permisos del usuario en el sistema.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
permission Enum	Es obligatorio y determina el tipo de permiso.
Los siguientes permisos de lectura regulan el acceso a cada módulo: • THEME_READ: Acceso y consulta al módulo Temas. • LEGEND_READ: Acceso y consulta al módulo Unidades. • SUGGESTION_READ: Acceso y consulta al módulo Sugerencias. • MISTAKE_READ: Acceso y consulta al módulo Erratas. • UPDATE_READ: Acceso y consulta al módulo Publicaciones. Los siguientes permisos de escritura otorgan las acciones dentro de cada módulo: • THEME_WRITE: Control total en el módulo Temas. • LEGEND_WRITE: Control total en el módulo Unidades. • SUGGESTION_WRITE: Control total en el módulo Sugerencias. • MISTAKE_WRITE: Control total en el módulo Erratas.	

Clase: HistoricalStatus	
Descripción: Clase abstracta que unifica los historiales de las diferentes entidades.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
creationDate Date	Es obligatorio y determina la fecha de creación del registro.

Clase: HistoricalThemeStatus extends HistoricalStatus	
Descripción: Almacena el cambio de estado de temas y conversaciones.	
Atributo	Descripción
status Enum	Es obligatorio e identifica el estado aplicado en el historial.
Los diferentes estados por los que puede pasar un tema o conversación son: • INACTIVE: La conversación o el tema se encuentra inactivo. • CANCEL: La conversación o el tema se encuentra cancelado. • PUBLISHED: La conversación o el tema se encuentra publicado en TellMe.	

Clase: HistoricalMistakeStatus extends HistoricalStatus	
Descripción: Almacena el cambio de estado de las erratas notificadas.	
Atributo	Descripción
status Enum	Es obligatorio e identifica el estado aplicado en el historial.
Los diferentes estados por los que puede pasar una notificación son: • NEW: La notificación se encuentra en estado nuevo. • ACCEPTED: La notificación ha sido aceptada. • REJECTED: La notificación ha sido rechazada.	

Clase: HistoricalLegendStatus extends HistoricalStatus	
Descripción: Almacena el cambio de estado de las unidades y leyendas.	
Atributo	Descripción
status Enum	Es obligatorio e identifica el estado aplicado en el historial.
Los diferentes estados por los que puede pasar una unidad o leyenda son: • INACTIVE: La leyenda o la unidad se encuentra inactiva. • CANCEL: : La leyenda o la unidad se encuentra cancelada. • PUBLISHED: : La leyenda o la unidad se encuentra publicada en TellMe.	

Clase: HistoricalSuggestionStatus extends HistoricalStatus	
Descripción: Almacena el cambio de estado de las sugerencias realizadas.	
Atributo	Descripción
status Enum	Es obligatorio e identifica el estado aplicado en el historial.
Los diferentes estados por los que puede pasar una sugerencia son: • NEW: La sugerencia se encuentra en estado nuevo. • UNDER_REVIEW: La sugerencia se encuentra en proceso de estudio. • ACCEPTED: La sugerencia ha sido aceptada. • REJECTED: La sugerencia ha sido rechazada.	

Clase: Numeration	
Descripción: Lleva el control de los códigos asignados a los datos.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
sequence Integer	Es obligatorio y lleva el contador de los registros asignados.
code Enum	Es obligatorio e indica la entidad a la cual se le asigna un código.
Los valores del enumerado determinan la entidad a la cual se asigna la numeración: • THEME: Código para identificar a los temas. • LEGEND: Código para identificar las unidades. • CONVERSATION: Código para identificar las conversaciones. • LEGEND_LINE: Código para identificar las leyendas. • CONVERSATION_LINE: Código para identificar las líneas del diálogo.	

Clase: Level	
Descripción: Entidad que agrupa los temas por niveles de dificultad.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
status Enum	Es obligatorio e identifica el grado de dificultad.
Los valores de este enumerado son: • BEGINNER: Identifica al nivel de Principiante. • INTERMEDIARE: Identifica al nivel Intermedio. • ADVANCED: Identifica al nivel de Avanzado.	

Clase: Answer	
Descripción: Clase abstracta que unifica las peticiones hechas desde TellMe.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
creationDate Date	Es obligatorio y determina la fecha de creación del registro.
client String	Es obligatorio e indica el nombre de usuario de TellMe.
description String	Es obligatorio e indica la petición hecha por el usuario TellMe.

Clase: Mistake extends Answer	
Descripción: Clase que permite completar la errata notificada desde TellMe.	
Atributo	Descripción
originalLine String	Es obligatoria e indica la línea que contiene la errata.
solution String	Se almacena la solución adoptada sobre la errata.
status Enum	Es obligatorio y determina el estado actual de la notificación.
Los valores del enumerado son similares a los de la clase HistoricalMistakeStatus.	

Clase: Suggestion extends Answer	
Descripción: Clase que permite completar la sugerencia realizada desde TellMe.	
Atributo	Descripción
status Enum	Es obligatorio y determina el estado actual de la solicitud.
Los valores del enumerado son similares a los de la clase HistoricalSuggestionStatus.	

Clase: Theme	
Descripción: Entidad que agrupa las conversaciones en temas o temáticas.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
description String	Es obligatoria y registra el título del tema o de la conversación.
code String	Identificador único que identifica al tema o conversación.
creationDate Date	Es obligatoria y registra la fecha de creación.
status Enum	Es obligatorio y marca el estado actual del tema o conversación.
Los valores del enumerado son similares a los de la clase HistoricalThemeStatus.	

Clase: ConversationLine	
Descripción: Entidad que almacena las líneas de cada diálogo.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
line String	Es obligatoria y registra la línea de la conversación.
orderLine Integer	Es obligatorio e indica la posición de la línea en el diálogo.
code String	Identificador único que identifica la línea del diálogo en la app.

Clase: Legend	
Descripción: Entidad que agrupa las leyendas en unidades.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
description String	Contiene el título de la unidad o el contenido de la leyenda. Es requerido.
creationDate Date	Es obligatoria y registra la fecha de creación.
code String	Identificador único que identifica a la unidad o leyenda.
status Enum	Es obligatorio e indica el estado actual de la unidad o leyenda.
Los valores del enumerado son similares a los de la clase HistoricalLegendStatus.	

Clase: Avatar	
Descripción: Clase que representa a los interlocutores de las conversaciones.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
code Enum	Es obligatorio e identifica al personaje animado.
Los valores de este enumerado identifican a los personajes de la aplicación móvil: • A1: Corresponde al personaje animado June, actriz principal de TellMe. • A2: Identifica a la actriz Lucy. • A3: Representa al interlocutor Timmy. • A4: Corresponde al personaje animado Paul.	

Clase: Update	
Descripción: Clase que registrará los contenidos didácticos publicados para TellMe.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
description String	Es obligatoria y almacena una copia del material publicado.
creationDate Date	Es obligatoria y registra la fecha de creación.
publishDate Date	Es obligatoria y registra la fecha GMT que se utilizará para la sincronización con la aplicación móvil.
type Enum	Es obligatorio e indica el tipo de publicación realizado.
Los valores de este enumerado son: • NEW: Indica que es la primera vez que se publica un contenido didáctico. • UPDATING: Indica que se ha realizado la actualización de un material ya publicado.	

Manual operativo de usuario de TellMe Server

En este manual, se ha detallado la funcionalidad que ofrece la aplicación. Además se ha completado con el diseño de la interfaz y con las explicaciones propias de un caso de uso.

TellMe Server es un sistema de información que permite al usuario generar y gestionar contenidos didácticos para la aplicación móvil TellMe. En este manual de usuario, se intentará detallar toda la funcionalidad que ofrece esta aplicación web.

Iniciando sesión en TellMe Server

El usuario deberá autenticarse ante el sistema con el nombre de usuario y contraseña que le haya proporcionado el administrador del sistema.



Figura 14. Inicio de sesión.

Tras pulsar “*Iniciar sesión*”, el sistema verificará su autenticidad y le asignará los permisos que el administrador le haya configurado. Por otro lado, el sistema podría prohibir el acceso al usuario en los siguientes casos:

- El nombre de usuario y/o contraseña son erróneos.
- El usuario carece de permisos para acceder a la aplicación.
- El usuario tiene inhabilitada su cuenta de usuario.

En estas situaciones, el usuario deberá contactar con el administrador del sistema para corregir la incidencia.

Navegando por TellMe Server

El usuario tiene a su disposición los siguientes módulos:



Figura 15. Menú de la aplicación.

- **Temas:** El usuario podrá generar y gestionar el contenido didáctico que se utilizará en la aplicación móvil. Para ello, el usuario podrá crear conversaciones que estarán organizadas por temas.
- **Unidades:** Con el objetivo de complementar las conversaciones creadas, el usuario podrá elaborar una serie de contenidos extra como, por ejemplo, aspectos gramaticales del idioma o hacer mención de vocabulario o expresiones relevantes. Dichos contenidos extras se denominan leyendas y están organizadas por unidades.
- **Sugerencias:** El usuario podrá atender y gestionar las peticiones realizadas por el usuario de TellMe para la creación de nuevos contenidos didácticos.
- **Erratas:** El usuario podrá atender y gestionar las erratas que el usuario de la aplicación móvil haya detectado en alguna conversación o leyenda.
- **Publicaciones:** Desde este módulo, el usuario podrá consultar el contenido didáctico que ha sido publicado para la aplicación móvil.
- **Quiénes somos:** Esta sección habla de Tecnología Social, grupo de investigación de la Universidad de Las Palmas de Gran Canaria.
- **Salir:** Con este enlace, el usuario podrá cerrar su sesión.

Módulo Temas

Este módulo permite buscar, crear y tratar los contenidos didácticos a los que tendrá acceso la aplicación TellMe.

- **Creación de temas**

Para crear un tema, el usuario deberá pulsar el botón “*Nuevo tema*” y el sistema abrirá el siguiente modal. En él, deberá indicar para qué nivel de dificultad corresponde el tema (Principiante, Intermedio o Avanzado) y el título del mismo. Si el usuario no visualiza dicho botón, esto indicará que no tiene los permisos correctos y deberá ponerse en contacto con el administrador del sistema.

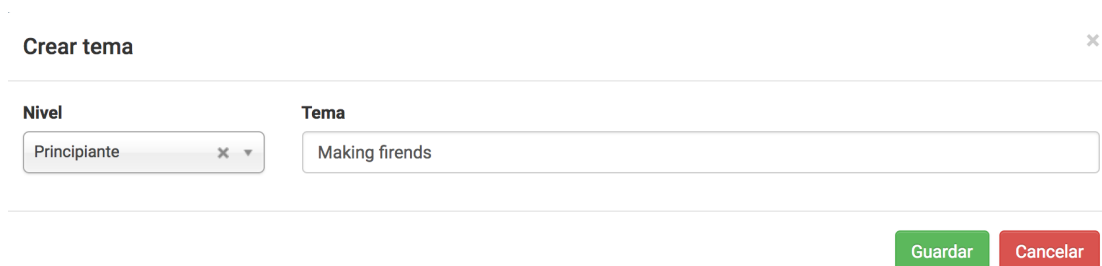


Figura 16. Creación de tema.

Tras rellenar los campos, el usuario deberá pulsar “*Guardar*” para que el sistema verifique los datos introducidos y cree el nuevo tema. El sistema informará del éxito de la creación con el siguiente mensaje “*El tema se ha guardado con éxito*”. En caso contrario, el sistema informará con el siguiente mensaje “*Ha ocurrido un error al guardar el tema*”.

Tras la creación, el nuevo tema estará en estado inactivo. En caso de no introducir los campos requeridos, el sistema resaltará los campos requeridos.

- **Búsqueda de temas**

Formulario de búsqueda de temas y conversaciones. El formulario contiene los siguientes elementos:

- Tema:** Campo de texto para la descripción del tema.
- Nivel:** Selector de nivel de dificultad (Principiante).
- Estado:** Selector de estado (Selecciona un estado...).
- Conversación:** Campo de texto para el título de la conversación.
- Botones:** BUSCAR (con icono de lupa) y LIMPIAR (con icono de borrar).

Figura 17. Buscador de temas/conversaciones.

Con esta herramienta, se podrá buscar un tema por los siguientes campos:

- Tema: Descripción del tema.
- Nivel: Nivel de dificultad del tema (Principiante, Intermedio o Avanzado).
- Estado: Estado en el que se encuentra actualmente el tema. Tipos:
 - Inactivo: El tema no es visible desde la aplicación móvil.
 - Cancelado: El tema ha sido cancelado y no es visible TellMe.
 - Publicado: El tema es visible desde la aplicación móvil.
- Conversación: Si un tema contuviera una o varias conversaciones, el usuario podría buscar por el título de la conversación.

Resultado del buscador de temas. La interfaz muestra un encabezado "Listado de temas" y un filtro "Mostrar 5". La tabla de resultados contiene la siguiente información:

Nivel	Tema	Nº Conversaciones	Creado	Estado	Acciones
Principiante	Making frends	0	09/05/2017	Inactivo	[Editar] [Eliminar]

Mostrando desde 1 hasta 1 de 1 registros totales. Paginación: 1.

Figura 18. Resultado del buscador de temas.

Tras cumplimentar los filtros, el usuario deberá pulsar “*Buscar*” para que el sistema recupere los temas que se ajusten a lo solicitado. Para borrar los filtros establecidos en el buscador, el usuario podrá pulsar el botón “*Limpiar*”. Como resultado de la búsqueda, el usuario consultará la siguiente información:

- Nivel: Nivel de dificultad asociado al tema.
- Tema: Título del tema.
- Nº conversaciones: Número de conversaciones que contiene el tema.
- Creado: Fecha de creación del tema.
- Estado: Estado actual del tema.
- Acciones: Describe las acciones que se pueden realizar sobre un tema:



Editar tema: El usuario podrá modificar aspectos como el nivel al que está asociado el tema o la descripción del mismo. Además, podrá crear nuevas conversaciones que compondrán el tema.



Consultar tema: El usuario podrá consultar las conversaciones asociadas al tema.



Cancelar tema: Si el usuario lo considera oportuno, podrá cancelar un tema. Esto implica que las conversaciones que contenga, también serán canceladas.

Según el estado del tema, el usuario tendrá disponible las acciones:

Estados	Inactivo	Cancelado	Publicado
Acciones	 		

- **Editar un tema**

Para la edición, el usuario deberá pulsar el icono correspondiente y el sistema navegará al detalle del tema. Si el usuario lo estima oportuno, podrá cambiar el nivel y la descripción del tema. Para aplicar las modificaciones, el usuario deberá pulsar “Actualizar” y el sistema le informará con el mensaje “El tema ha sido guardado con éxito”. En caso contrario, informará con “Ha ocurrido un error al guardar el tema”.

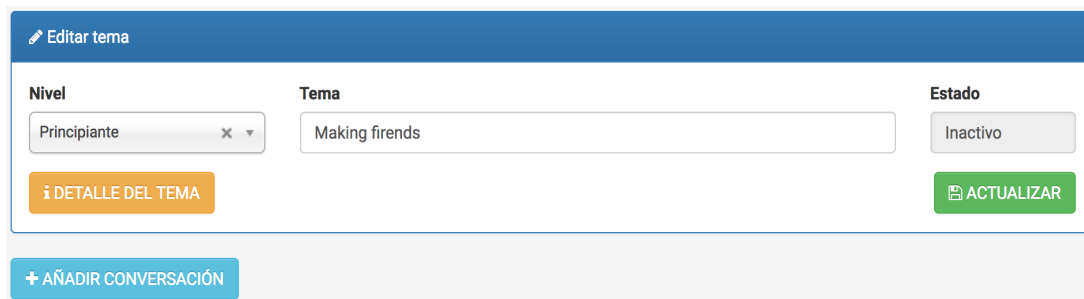


Figura 19. Editar tema.

Si el tema estuviera Publicado, el sistema no permitirá modificar el nivel, ya que esto implicaría, en la aplicación móvil, trasladar los contenidos didácticos de este tema de un nivel a otro. Además, el usuario podrá consultar el historial de estados y publicaciones que haya sufrido el tema.

Historial de estados		
Fecha	Usuario	Estado
22/10/2016 20:33:18	admin	Publicado
22/10/2016 20:32:45	admin	Inactivo

Historial de publicaciones					
Publicación	Creado	Publicado	Tipo	Elemento	Usuario
At school	22/10/2016 20:40:45	22/10/2016 19:40:45	Actualización	Tema	admin
At the school	22/10/2016 20:33:18	22/10/2016 19:33:18	Nuevo	Tema	admin

Figura 20. Historial de estados y publicaciones del tema.

En este caso, se puede observar que el usuario admin publicó el tema el día 22/10/2016 a las 19:33:18. Esto provocó que se generara una nueva publicación para que el tema fuera visible para la aplicación móvil.

Para poder llevar una sincronización entre este sistema y la aplicación móvil, se debe adaptar la fecha de publicación a la franja horaria GMT. De esta manera, los usuarios de la aplicación móvil que se encuentren en Rusia, Francia o México no tendrán problemas para consultar los contenidos didácticos publicados.

Además, tras la publicación, dicho tema ha sufrido una actualización en su título. Con la ayuda de la sincronización, la aplicación móvil TellMe podrá detectar los cambios y actualizarlos en su base de datos.

Finalmente, el usuario cuenta con una herramienta para crear diálogos dentro del tema. Para más información, consulte el apartado “*Añadir conversación*”.

- **Consultar un tema**

Para consultar, el usuario deberá pulsar el icono correspondiente y el sistema navegará al detalle del tema. En este caso, el usuario no podrá modificar el nivel ni la descripción del tema, así como tampoco podrá crear nuevas conversaciones.

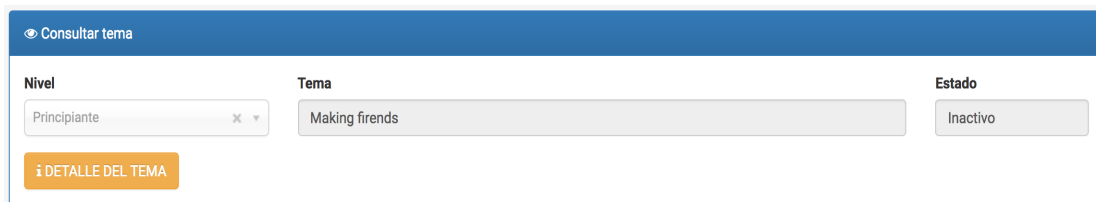


Figura 21. Consultar un tema.

- **Cancelar un tema**

Para cancelar, el usuario deberá pulsar el icono correspondiente y el sistema mostrará el siguiente mensaje de confirmación.

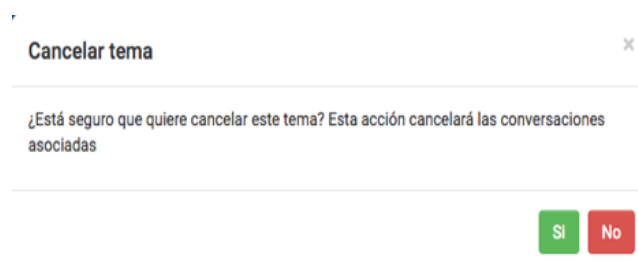


Figura 22. Cancelar un tema.

Si el usuario pulsa “Si”, el sistema cancelará el tema y las conversaciones que contenga e informará al usuario con el mensaje “*Se ha cancelado el tema*”. Por tanto, este bloque de contenidos didácticos no podrán ser visibles desde la aplicación móvil.

- **Añadir una conversación**

Siguiendo con el ejemplo del tema “*Making friends*”, el usuario deberá pulsar el botón “*Añadir conversación*”.

Figura 23. Añadir conversación.

Tras introducir el título de la conversación, el usuario deberá pulsar “*Guardar*” y el sistema informará del éxito con “*La conversación ha sido creada con éxito*”. En caso contrario, con “*Ha ocurrido un error al crear la conversación*”. Una vez creada la conversación Inactiva, el usuario podrá consultar todas las conversaciones asociadas al tema “*Making friends*”.

Conversación	Leyendas	Líneas	Actores	Estado	Creado	Acciones
Presentation		0	0	Inactivo	09/05/2017	

Figura 24. Listado de conversaciones.

La información es presentada con los siguientes campos:

- **Conversación:** Título de la conversación.
- **Leyendas:** Tooltip que muestra las leyendas usadas en la conversación.
- **Líneas:** Número de líneas que contiene el diálogo.
- **Actores:** Número de personajes utilizados en el diálogo.
- **Estado:** Estado actual de la conversación.
- **Creado:** Fecha de creación de la conversación.
- **Acciones:** Describe las acciones que se pueden realizar:



Editar conversación: El usuario podrá navegar al detalle de la conversación y modificar su descripción. Además, podrá añadir las líneas del diálogo.



Consultar conversación: El usuario podrá navegar al detalle de la conversación y consultar las líneas del diálogo.



Cancelar conversación: Si el usuario lo considera oportuno, podrá cancelar una conversación. Esto implica que no será visible para la aplicación móvil.



Publicar conversación: Cuando el usuario haya redactado y revisado los contenidos didácticos de la conversación, podrá publicarla. De esta manera, la aplicación móvil tendrá visibilidad sobre la conversación.

Según el estado de la conversación, el usuario tendrá disponible las acciones:

Estados	Inactivo	Cancelado	Publicado
Acciones	  		

- **Editar una conversación**

El usuario deberá pulsar en el icono correspondiente y el sistema navegará al detalle de la conversación. Si lo considera oportuno, podrá modificar el título de la conversación cuando, por ejemplo, considere que el título de la conversación no se ajusta al diálogo o exista alguna errata.

Para aplicar los cambios, el usuario deberá pulsar “Actualizar” y el sistema informará del éxito con el mensaje “La conversación ha sido actualizada con éxito”. Por lo contrario, informará con “Ha ocurrido un error al actualizar la conversación”.

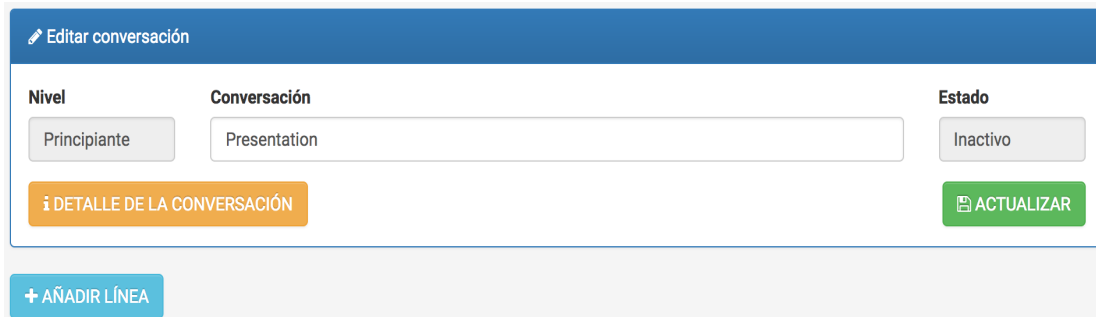


Figura 25. Editar una conversación.

Además, el usuario podrá consultar el historial de estados y publicaciones que haya sufrido la conversación.

Historial de estados					
Fecha	Usuario		Estado		
23/10/2016 19:28:27	admin		Publicado		
23/10/2016 19:27:04	admin		Inactivo		

Historial de publicaciones					
Publicación	Creado	Publicado	Tipo	Elemento	Usuario
Mommy, Can we play outside?	23/10/2016 19:31:16	23/10/2016 18:31:16	Actualización	Línea de conversación	admin
It's rain!	23/10/2016 19:28:27	23/10/2016 18:28:27	Nuevo	Conversación	admin

Figura 26. Historial de estados y publicaciones de la conversación.

En este caso, se puede observar que el usuario admin publicó la conversación el día 23/10/2016 a las 18:28:27. Esto provocó que se generara una nueva publicación para que el diálogo fuera visible para la aplicación móvil. Además, tras la publicación, una de sus líneas sufrió una actualización.

Por otro lado, mientras la conversación está inactiva, el usuario podrá añadir las líneas que compondrán el diálogo “*Presentation*”. Para ello, deberá pulsar “*Añadir línea*”. Para más información, consulte el apartado “*Añadir una línea a la conversación*”. Si la conversación estuviera Publicado, el sistema no permitirá añadir líneas al diálogo.

Finalmente, si la conversación contuviera líneas, el usuario podrá visualizarlas y realizar acciones como editarlas o cancelarlas. Estas acciones se detallarán más adelante en las secciones “*Editar línea de conversación*” y “*Eliminar línea de conversación*” respectivamente.

- **Consultar una conversación**

Para consultar, el usuario deberá pulsar el icono correspondiente y el sistema navegará al detalle de la conversación. En este caso, el usuario no podrá modificar la descripción de la conversación y tampoco podrá añadir nuevas líneas, editarlas o cancelarlas.



Figura 27. Consultar una conversación.

- **Cancelar una conversación**

Si se considera oportuno, el usuario podrá cancelar una conversación. Para ello, deberá pulsar el icono correspondiente sobre la conversación que desee cancelar y el sistema mostrará el siguiente mensaje de confirmación.

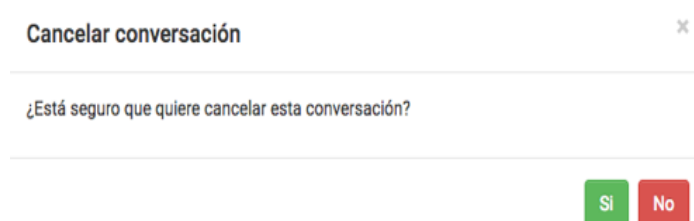


Figura 28. Cancelar una conversación.

Si el usuario pulsa “*Si*”, el sistema cancelará la conversación. Por tanto, no podrá ser visible desde la aplicación móvil. El sistema informará del éxito con el mensaje “*Se ha cancelado la conversación*”. Por lo contrario, informará con “*Ha ocurrido un error al cancelar la conversación*”.

- **Publicar una conversación**

Cuando el usuario haya redactado y revisado el diálogo, podrá hacerla visible en la aplicación móvil. Esto se denomina “**publicar**”. Para ello, deberá pulsar el botón correspondiente y el sistema mostrará el siguiente mensaje de confirmación.

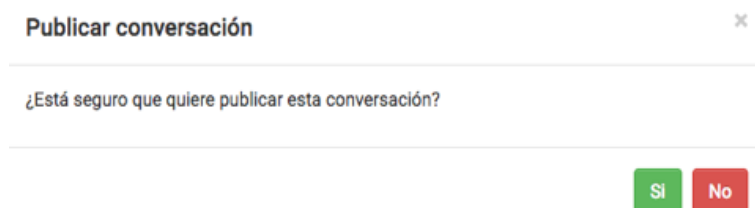


Figura 29. Publicar una conversación.

Si el usuario pulsa “*Si*”, el sistema verificará que dicha conversación tiene al menos cuatro líneas de diálogo. En caso contrario, el sistema informará con el mensaje “*La conversación debe contener al menos 4 líneas*”. Esta restricción evitará publicar conversaciones con poco contenido didáctico.

Tras superar la restricción, el diálogo es visible para la aplicación móvil. Además, si el tema que contiene dicha conversación estuviera Inactivo, el sistema también lo haría visible, actualizando el estado del tema a Publicado.

Asimismo, si en la redacción del diálogo se utilizó alguna leyenda, el sistema también la haría visible, actualizando su estado a Publicado. Finalmente, si la unidad que contiene a la leyenda estuviera también Inactiva, el sistema también la haría visible. Los conceptos de “*Leyendas*” y “*Unidades*” se entrarán en detalle más adelante.

Por tanto, al publicar, el sistema hará visible para la aplicación móvil los contenidos didácticos descritos en el tema, la conversación, las leyendas utilizadas y sus unidades.

- **Añadir una línea a la conversación**

Siguiendo con el ejemplo de la conversación “*Presentation*” asociado al tema “*Making friends*”, el usuario deberá acceder al detalle del diálogo para poder añadir nuevas las líneas. Para ello, deberá pulsar el botón “*Añadir línea*” y el sistema mostrará la siguiente ventana.

Figura 30. Añadir una línea a la conversación.

El usuario deberá cumplimentar la línea del diálogo, así como el actor que la interpretará en la aplicación móvil. Si se considera oportuno, se puede complementar con leyendas. Para ello, deberá buscar la unidad a la que pertenece la leyenda y asociarla. Más adelante se explicará la creación y usos de las unidades y leyendas.

Para aplicar los cambios, se deberá pulsar “*Guardar*” y el sistema verificará los campos requeridos y podrá crear la línea del diálogo. El sistema informará del éxito de la creación con el mensaje “*La línea ha sido creada con éxito*”. En caso contrario, el sistema informará con el mensaje “*No se ha podido crear la línea*”. Si no se introducen los campos requeridos, el sistema resaltará cuales son los campos obligatorios. De esta forma, tendremos el siguiente diálogo:

Detalle de la conversación				
Orden	Actor	Línea de la conversación ☆ Leyenda	Acciones	
1		Hello, I'm June. ☆ 'Hi' or 'Hello' is an informal way to exchange greetings.		
2		Hi, I'm Timmy.		
3		Nice to meet you Timmy. ☆ You can use 'nice to meet you' when you meet someone for the first time.		
4		Nice to meet you too.		
5		How old are you?		
6		I'm seven years old, and you?		
7		I'm eight years old.		
8		I'm sorry but I have to go, goodbye.		
9		Bye.		

Figura 31. Ejemplo de conversación.

Por tanto, y para completar este ejemplo, la conversación “*Presentation*” estará compuesta por nueve líneas que serán interpretadas por los actores June y Timmy.

Asimismo, se han añadido dos leyendas que ayudarán a asimilar, reforzar y enriquecer el aprendizaje del usuario de la aplicación móvil. Por otro lado, el sistema muestra las opciones de edición y eliminación en cada línea del diálogo. Estas acciones se detallarán a continuación.

- **Editar una línea de conversación.**

El usuario podrá modificar una línea con el objetivo de adaptar su contenido didáctico al contexto. Para ello, deberá pulsar el botón “*Editar línea*” y el sistema abrirá una ventana con la información de la línea.

Figura 32. Editar una línea de conversación.

Mientras la conversación se encuentre en estado Inactivo, el usuario podrá modificar todos los datos. En el momento que dicha conversación sea publicada, solo se podrá modificar la línea del diálogo.

Para aplicar cambios, se deberá pulsar “*Guardar*” y el sistema informará del éxito con el mensaje “*La línea ha sido actualizada con éxito*”. En caso contrario, informará con el mensaje “*Ha ocurrido un error al actualizar la línea*”.

- **Eliminar una línea de conversación**

Mientras la conversación se encuentre Inactiva, el usuario podrá eliminar líneas con el fin de quitar contenidos didácticos que no utilizará en este diálogo. Si el usuario lo considera oportuno, deberá pulsar el botón “*Eliminar línea*” y el sistema le pedirá que confirme la acción.

Figura 33. Solicitud de eliminación de una línea del diálogo.

Siguiendo con el ejemplo de la conversación “*Presentation*”, se eliminarán las líneas 5, 6 y 7 ya que no están directamente relacionadas con los saludos.

5		How old are you?	 
6		I'm seven years old, and you?	 
7		I'm eight years old.	 

Figura 34. Eliminación de varias líneas del diálogo.

Tras la eliminación, el sistema ordenará las demás líneas del diálogo para evitar saltos. Por tanto, la conversación final estará compuesta por:

Detalle de la conversación			
Orden	Actor	Línea de la conversación ☆ Leyenda	Acciones
1		Hello, I'm June. ☆ 'Hi' or 'Hello' is an informal way to exchange greetings.	 
2		Hi, I'm Timmy.	 
3		Nice to meet you Timmy. ☆ You can use 'nice to meet you' when you meet someone for the first time.	 
4		Nice to meet you too.	 
5		I'm sorry but I have to go, goodbye.	 
6		Bye.	 

Figura 35. Conversación resultante de la eliminación de varias líneas del diálogo.

Módulo Unidades

En este módulo, el usuario podrá crear contenidos didácticos extra que podrá asociar a las líneas de una conversación. De esta manera, el diálogo se verá reforzado y enriquecido.

- **Creación de unidades**

Para crear una unidad, se deberá pulsar el botón “*Nueva unidad*” y el sistema abrirá el siguiente modal. En él, el usuario deberá indicar el título de la unidad.

Crear unidad

Unidad

Greetings and Farewells

Guardar

Cancelar

Figura 36. Crear una unidad.

Tras cumplimentar el campo requerido, el usuario deberá pulsar “*Guardar*” para que el sistema lo verifique y cree la nueva unidad. Éste informará con el mensaje “*La unidad ha sido guardada con éxito*”. En caso contrario, el sistema informará con el mensaje “*Ha ocurrido un error al guardar la unidad*”. En el caso de no introducir el campo requerido, el sistema lo resaltará como obligatorio.

- **Búsqueda de unidades**

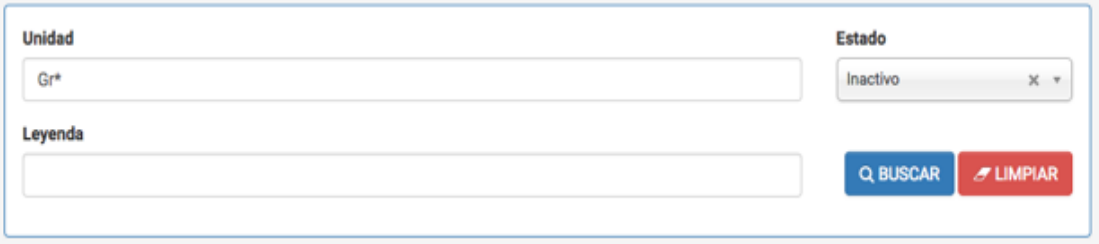
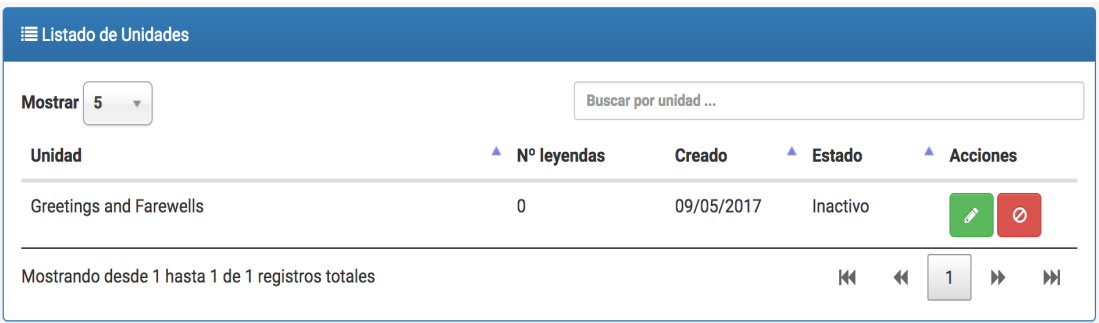


Figura 37. Buscador de unidades/leyendas.

Con esta herramienta, el usuario podrá buscar una unidad por los siguientes campos:

- **Unidad:** Descripción la unidad didáctica.
- **Estado:** Estado en el que se encuentra actualmente la unidad. Tipos:
 - Inactivo: La unidad no es visible desde la aplicación móvil.
 - Cancelado: La unidad ha sido cancelada y no es visible.
 - Publicado: La unidad es visible desde la aplicación móvil.
- **Leyenda:** Si una unidad contuviera una o varias leyendas, el usuario podría buscarlas por su descripción.

Tras cumplimentar los filtros de búsqueda, el usuario deberá pulsar el botón “*Buscar*” para que el sistema recupere las unidades que se ajusten a lo solicitado. Para borrar los filtros establecidos, el usuario podrá pulsar el botón “*Limpiar*”.



Unidad	Nº leyendas	Creado	Estado	Acciones
Greetings and Farewells	0	09/05/2017	Inactivo	 

Mostrando desde 1 hasta 1 de 1 registros totales

Figura 38. Resultado del buscador de unidades.

Como resultado, el usuario visualizará la siguiente información:

- **Unidad:** Título de la unidad.
- **Nº Leyendas:** Número de leyendas que contiene la unidad.
- **Creado:** Fecha de creación de la unidad.
- **Estado:** Estado actual de la unidad.
- **Acciones:** Describe las acciones que se pueden realizar sobre una unidad:



Editar unidad: El usuario podrá navegar al detalle de una unidad y podrá modificarla. Además podrá crear leyendas que utilizará en los diálogos.



Consultar unidad: El usuario podrá navegar al detalle de una unidad y consultar las leyendas asociadas.



Cancelar unidad: Si el usuario lo considera oportuno, podrá cancelar una unidad. Esto implica que sus leyendas, también serán canceladas.

Según el estado de la unidad, se habilitarán las siguientes opciones:

Estados	Inactivo	Cancelado	Publicado
Acciones	 		

- **Editar una unidad**

Para la edición, el usuario deberá pulsar el icono correspondiente y el sistema navegará al detalle de la unidad. Si el usuario lo considera oportuno, podrá cambiar la descripción de la unidad.

Para aplicar los cambios, se deberá pulsar “*Actualizar*”. Como consecuencia, el sistema informará con el mensaje “*La unidad ha sido guardada con éxito*”. En caso contrario, informará con el mensaje “*Ha ocurrido un error al guardar la unidad*”.

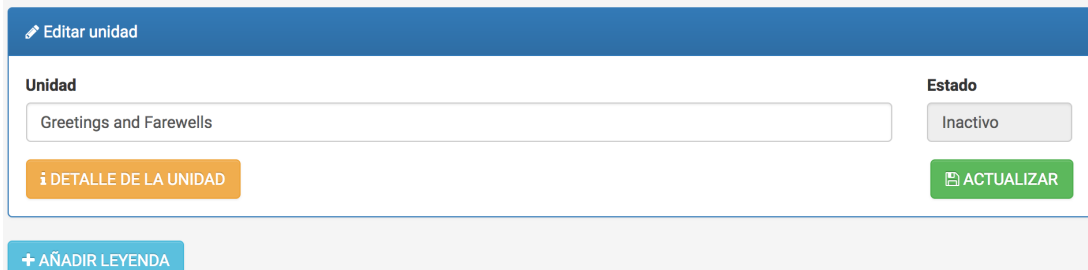


Figura 39. Edición de una unidad.

Además, el usuario podrá consultar el historial de estados y publicaciones que ha sufrido la unidad.

Historial de estados					
Fecha	Usuario		Estado		
24/10/2016 21:01:01	admin		Publicado		
24/10/2016 11:56:46	admin		Inactivo		

Historial de publicaciones					
Publicación	Creado	Publicado	Tipo	Elemento	Usuario
Greetings and Farewells	24/10/2016 21:01:01	24/10/2016 20:01:01	Nuevo	Unidad	admin

Figura 40. Historial de estados y publicaciones de una unidad.

En este caso, se puede observar que el usuario admin publicó la unidad el día 24/10/2016 a las 20:01:01. Esto provocó que se generara una nueva publicación para que la unidad fuera visible para la aplicación móvil.

Por otro lado, el usuario cuenta con una herramienta para crear leyendas dentro de la unidad. Para más información, consulte el apartado “*Añadir leyenda*”.

- **Consultar una unidad**

Para consultar una unidad, el usuario deberá pulsar el icono correspondiente y el sistema navegará al detalle del mismo. En este caso, el usuario no podrá modificar el título de la unidad y tampoco podrá crear nuevas leyendas.



Figura 41. Consultar una unidad.

- **Cancelar una unidad**

Para cancelar, el usuario deberá pulsar el icono correspondiente y el sistema mostrará el siguiente mensaje de confirmación.

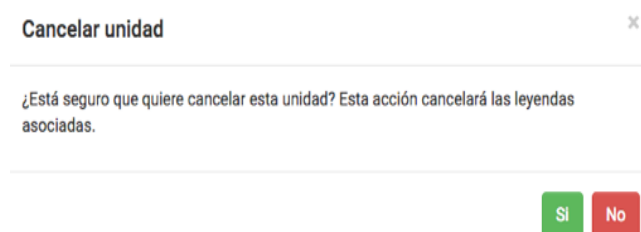


Figura 42. Cancelación de una unidad.

Si el usuario pulsa “Si”, el sistema cancelará la unidad y las leyendas que contenga e informará al usuario con el mensaje “*Se ha cancelado la unidad*”. Por tanto, este bloque de contenidos didácticos extra no serán visibles para TellMe.

Por otro lado, si una de las leyendas de esta unidad estuviera asociada a una línea de conversación Inactiva, el sistema no podrá cancelar la unidad. En este caso, el sistema informará con el mensaje “*La unidad cuenta con leyendas asociadas a las conversaciones*”. El usuario deberá editar las líneas involucradas y quitar la leyenda.

- **Añadir una leyenda**

Siguiendo con el ejemplo de la unidad “*Greetings and Farewells*”, el usuario deberá pulsar el botón “*Añadir leyenda*”.

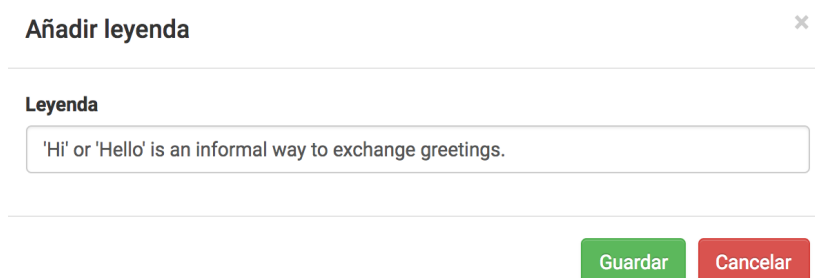


Figura 43. Añadir una leyenda a la unidad.

Tras cumplimentar la leyenda, el usuario deberá pulsar el botón “*Guardar*”. El sistema informará del éxito con el mensaje “*La leyenda ha sido creada con éxito*”. En

caso contrario, informará con el mensaje “*Ha ocurrido un error al crear la leyenda*”. Además podrá consultar el resto de leyendas de la unidad.

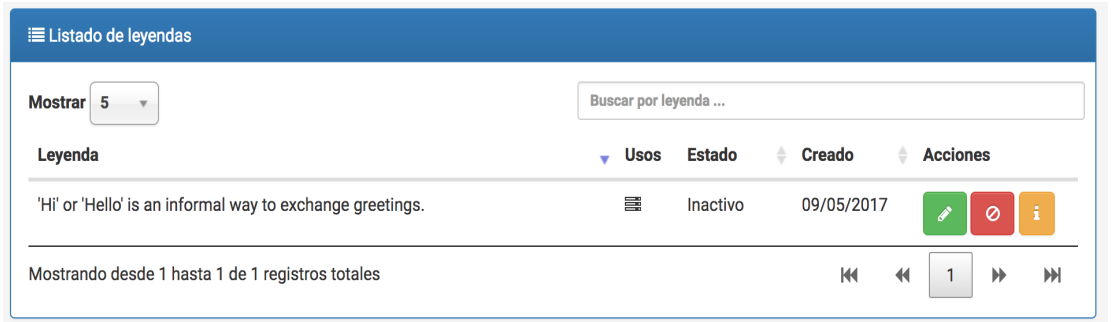


Figura 44. Listado de leyendas.

Dicha información es presentada con los siguientes campos:

- **Leyenda:** Descripción de la leyenda.
- **Usos:** Tooltip que muestra en qué conversaciones ha sido utilizada la leyenda.
- **Estado:** Estado actual de la leyenda.
- **Creado:** Fecha de creación de la leyenda.
- **Acciones:** Describe las acciones que se pueden realizar:

 **Editar leyenda:** El usuario podrá modificar la descripción de la leyenda.

 **Cancelar leyenda:** Si el usuario lo considera oportuno, podrá cancelar una leyenda. Esto implica que no será visible para la aplicación móvil.

 **Detalle de una leyenda:** Desde esta opción, el usuario podrá consultar el historial de estados por los que ha pasado una leyenda y su historial de publicaciones.

Según el estado de la leyenda, el usuario tendrá disponible las acciones:

Estados	Inactivo	Cancelado	Publicado
Acciones	  		 

- **Editar una leyenda**

Tras localizar la leyenda “*Hi' or 'Hello' is an informal way to exchange greetings.*” en la unidad “*Greetings and Farewells*”, el usuario podrá editarla. Para ello, deberá pulsar el botón correspondiente y el sistema abrirá la siguiente ventana.

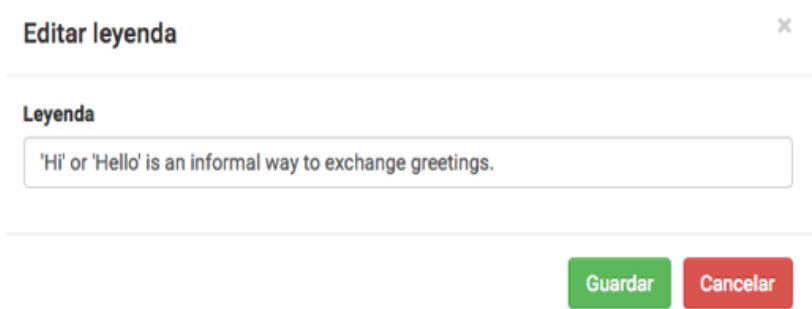


Figura 45. Editar una leyenda.

Tras realizar las modificaciones necesarias, el usuario deberá pulsar “Guardar” y el sistema informará del éxito con el mensaje “La leyenda ha sido actualizada con éxito”. En caso contrario, informará con el mensaje “Ha ocurrido un error al actualizar la leyenda”.

- **Cancelar una leyenda**

Para cancelar, el usuario deberá pulsar el icono y el sistema mostrará el siguiente mensaje de confirmación:

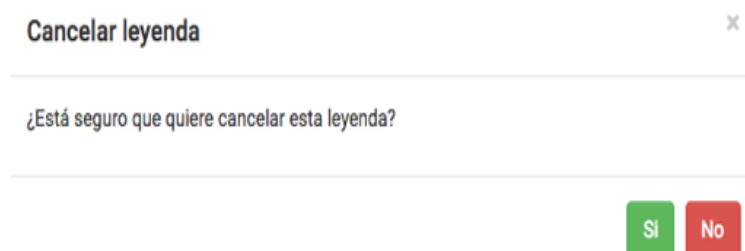


Figura 46. Cancelar una leyenda.

Si el usuario pulsa “Si”, el sistema cancelará la leyenda pero, si estuviera asociada a alguna conversación inactiva, el sistema no podrá cancelarla e informará con el mensaje “No se ha podido cancelar la leyenda porque está asociado a conversaciones inactivas”. En este caso, el usuario deberá editar las líneas involucradas y eliminar la leyenda.

- **Detalle de una leyenda**

A través de esta opción, el usuario podrá consultar el historial de estados por los que ha pasado una leyenda y su historial de publicaciones.

Historial de estados					
Fecha	Usuario		Estado		
24/10/2016 21:01:01	admin		Publicado		
24/10/2016 12:11:08	admin		Inactivo		

Historial de publicaciones					
Publicación	Creado	Publicado	Tipo	Elemento	Usuario
'Hi' or 'Hello' is an informal way to exchange greetings.	25/10/2016 11:58:19	25/10/2016 10:58:19	Actualización	Leyenda	admin
'Hi' or 'Hello' is an informal way to exchange greetings.	24/10/2016 21:01:01	24/10/2016 20:01:01	Nuevo	Leyenda	admin

Figura 47. Historial de estados y publicaciones de una leyenda.

En este caso, se puede observar que el usuario admin publicó la leyenda el día 24/10/2016 a las 20:01:01. Esto provocó que se generara una nueva publicación para

que la leyenda fuera visible para la aplicación móvil. Además, tras la publicación, la leyenda sufrió una actualización.

Módulo de Sugerencias

A través de este módulo, el usuario podrá atender y gestionar las peticiones que realice el usuario de TellMe. Para ello, el usuario cuenta con varias herramientas, como un buscador de sugerencias y la gestión de las mismas.

Formulario de búsqueda de sugerencias. El formulario contiene los siguientes elementos:

- Estado:** Un menú desplegable con la opción "Selecciona un estado...".
- Sugerencia:** Un campo de texto para ingresar la descripción de la sugerencia.
- Fecha creado desde:** Un campo de fecha con el valor "24/10/2016" y un icono de calendario.
- Fecha creado hasta:** Un campo de fecha con el valor "30/10/2016" y un icono de calendario.
- Botones:** Un botón azul "BÚSCAR" y un botón rojo "LIMPIAR".

Figura 48. Buscador de sugerencias.

El usuario podrá buscar una sugerencia a través de los siguientes campos:

- **Estado:** Estado en el que se encuentra actualmente la sugerencia. Tipos:
 - Nuevo: La sugerencia no ha sido atendida.
 - En estudio: La sugerencia está siendo atendida.
 - Aceptado: La sugerencia ha sido aceptada y el usuario creará contenidos didácticos que se ajusten a lo solicitado.
 - Rechazado: La sugerencia ha sido rechazada y, por tanto, el usuario no creará contenidos didácticos.
- **Sugerencia:** Descripción de la sugerencia.
- **Fecha creado desde:** Rango inferior de la fecha de creación de la sugerencia.
- **Fecha creado hasta:** Rango superior de la fecha de creación de la sugerencia.

Tras cumplimentar los filtros de búsqueda, el usuario deberá pulsar el botón “*Buscar*” para que el sistema recupere las sugerencias que se ajusten a lo solicitados. Para borrar los filtros establecidos, el usuario podrá pulsar el botón “*Limpiar*”.

Tabla de resultados de la búsqueda de sugerencias. La tabla muestra la siguiente información:

Id	Sugerencia	Creado	Cliente	Estado	Acciones
1	me gustaría que hubieran conversaciones sobre act...	30/05/2017	snaranjo	Nuevo	[Iconos de acciones]

Mostrando desde 1 hasta 1 de 1 registros totales

Figura 49. Resultados del buscador.

Como resultado, el usuario visualizará la siguiente información:

- **Id:** Número de sugerencia registrada en el sistema.
- **Sugerencia:** Petición realizada por el usuario de la aplicación móvil.
- **Creado:** Fecha de la petición.
- **Usuario:** Usuario de la aplicación móvil que ha realizado la petición.
- **Estado:** Estado actual en la que se encuentra la sugerencia.
- **Acciones:** Describe las acciones que se pueden realizar:

-  **Estudiar sugerencia:** Cuando el usuario se disponga a atender una petición, podrá marcarla con este estado, para diferenciarla de las no atendidas.
-  **Aceptar sugerencia:** Tras el estudio de una sugerencia y su correspondiente creación de contenidos didácticos, el usuario podrá marcar la petición como aceptada.
-  **Rechazar sugerencia:** Si el usuario lo considera oportuno, podrá rechazar peticiones ya que porque la sugerencia está duplicada o el contenido de la misma es inadecuada.
-  **Detalle de la sugerencia:** Desde esta opción, el usuario podrá consultar el historial de estados por los que ha pasado una sugerencia.

Según el estado de la sugerencia, el usuario tendrá disponible las acciones:

Estados	Nuevo	En estudio	Aceptado	Rechazado
Acciones	   	  		

- **Estudiar una sugerencia**

Cuando el usuario de TellMe sugiere nuevos contenidos didácticos, éstos se crean en estado “*Nuevo*”. Desde este punto de partida, el usuario podrá atender la petición. Para ello, la marcará como “*En estudio*” para diferenciarlas de aquellas que aún no han sido atendidas.



Figura 50. Estudiar una sugerencia.

Si el usuario pulsa “*Si*”, el sistema marcará la sugerencia como “*En estudio*” e informará del éxito con el mensaje “*La sugerencia ha sido revisada con éxito*”. En caso contrario, informará con “*Ha ocurrido un error al revisar la sugerencia*”.

- **Aceptar una sugerencia**

Tras el estudio de una sugerencia y su correspondiente creación de contenidos didácticos, el usuario podrá marcar la petición como aceptada para darla como finalizada. Para ello, deberá pulsar el botón correspondiente y el sistema le solicitará confirmación.

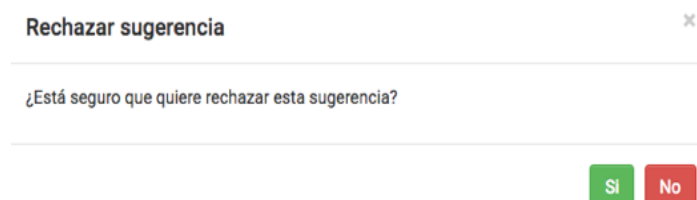


Figura 51. Aceptación de una sugerencia.

Si el usuario pulsa “Sí”, el sistema marcará la sugerencia como “Aceptada” e informará del éxito con el mensaje “La sugerencia ha sido aceptada con éxito”. En caso contrario, informará con “Ha ocurrido un error al aceptar la sugerencia”.

- **Rechazar una sugerencia**

Por otro lado, si el usuario lo considera oportuno, podrá rechazar peticiones ya que, por ejemplo, la sugerencia está duplicada o su contenido es inadecuado. Para ello, deberá pulsar el botón correspondiente y el sistema le solicitará confirmación.



Rechazar sugerencia

¿Está seguro que quiere rechazar esta sugerencia?

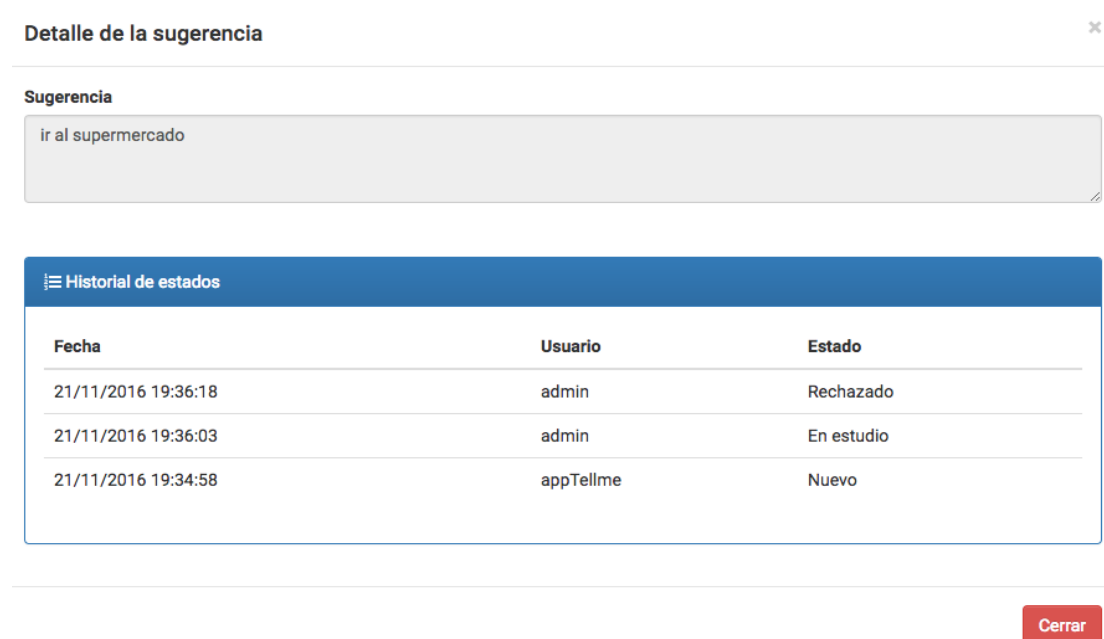
Si No

Figura 52. Cancelación de una sugerencia.

Si el usuario pulsa “Sí”, el sistema marcará la sugerencia como “Rechazada” e informará del éxito con el mensaje “La sugerencia ha sido rechazada con éxito”. En caso contrario, informará con “Ha ocurrido un error al rechazar la sugerencia”.

- **Detalle de una sugerencia**

A través de esta opción, el usuario podrá consultar el historial de estados por los que ha pasado una sugerencia. En este caso, la sugerencia ha sido rechazada.



Detalle de la sugerencia

Sugerencia

ir al supermercado

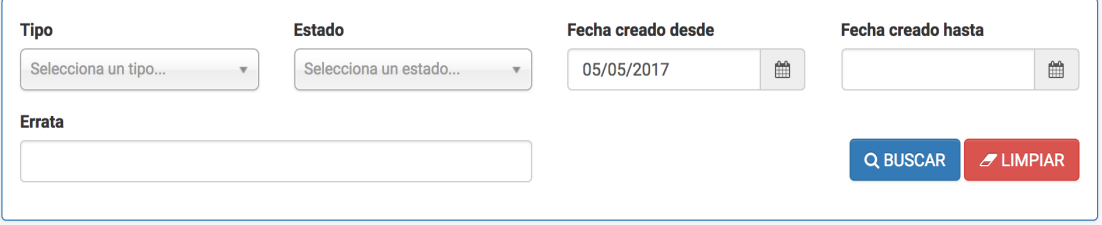
Historial de estados		
Fecha	Usuario	Estado
21/11/2016 19:36:18	admin	Rechazado
21/11/2016 19:36:03	admin	En estudio
21/11/2016 19:34:58	appTellme	Nuevo

Cerrar

Figura 53. Detalle de una sugerencia.

Módulo de Erratas

A través de este módulo, el usuario podrá atender y gestionar las notificaciones de erratas que realice el usuario de TellMe. Para ello, el usuario cuenta con varias herramientas, como un buscador de erratas y la gestión de las mismas.



Formulario de búsqueda de erratas. Incluye campos para Tipo, Estado, Fecha creado desde, Fecha creado hasta, y un campo de texto para la Errata. Botones de BUSCAR y LIMPIAR.

Figura 54. Buscador de erratas.

El usuario podrá buscar erratas por medio de los siguientes campos:

- Tipo: Identifica el elemento con la errata.
- Estado: Estado en la que se encuentra la errata. Tipos:
 - Nuevo: La errata no ha sido atendida.
 - Aceptada: La errata ha sido aceptada y se han corregido los datos.
 - Rechazada: La errata ha sido rechazada.
- Fecha creación desde: Rango inferior de la fecha de creación de la errata.
- Fecha creación hasta: Rango superior de la fecha de creación de la errata.

Tras cumplimentar los filtros de búsqueda, el usuario deberá pulsar el botón "Buscar" para que el sistema recupere las erratas que se ajusten a lo solicitado. Para borrar los filtros establecidos, el usuario podrá pulsar el botón "Limpiar".

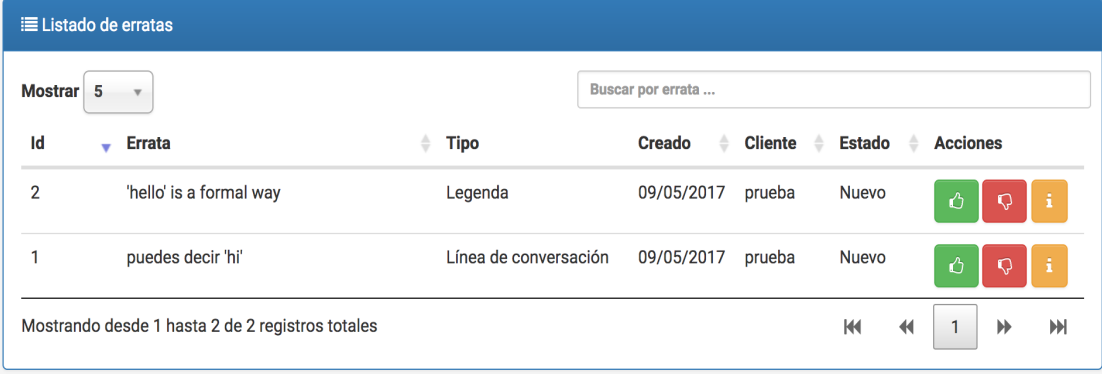


Tabla de resultados de búsqueda de erratas. Muestra una lista de erratas con columnas: Id, Errata, Tipo, Creado, Cliente, Estado, y Acciones. Incluye botones de búsqueda y limpieza.

Id	Errata	Tipo	Creado	Cliente	Estado	Acciones
2	'hello' is a formal way	Legenda	09/05/2017	prueba	Nuevo	  
1	puedes decir 'hi'	Línea de conversación	09/05/2017	prueba	Nuevo	  

Mostrando desde 1 hasta 2 de 2 registros totales

Figura 55. Resultados del buscador.

Como resultado, el usuario visualizará la siguiente información:

- Id: Número de errata registrada en el sistema.
- Errata: Descripción del problema.
- Tipo: Elemento, línea de conversación o leyenda afectada.
- Creado: Fecha de registro del problema.
- Cliente: Nombre del usuario de TellMe que ha reportado el error.
- Estado: Estado actual de la notificación.
- Acciones: Describe las acciones que se pueden realizar:



Aceptar una errata: Si el usuario considera que es correcto lo que sugiere el usuario de TellMe, podrá aceptarla, modificando el contenido didáctico.



Rechazar una errata: Si el usuario considera que es incorrecto lo que sugiere el usuario de TellMe, podrá rechazarla.



Detalle de la errata: Desde esta opción, el usuario podrá consultar el historial de estados por los que ha pasado una errata.

Según el estado de la errata, el usuario tendrá disponible las acciones:

Estados	Nuevo			Aceptado	Rechazado
Acciones					

- **Aceptar una errata**

Tras analizar la errata detectada por un usuario de la aplicación móvil, el usuario podrá marcar la petición como aceptada para darla como finalizada. Para ello, deberá pulsar el botón correspondiente y el sistema le abrirá el siguiente modal para corregir la leyenda o la línea de la conversación afectada.

Editar leyenda

Leyenda

'Hi' is an informal way to exchange greetings.

Guardar

Cancelar

Figura 56. Aceptación de una errata de una leyenda.

Si el usuario pulsa “*Guardar*”, el sistema marcará la errata como “*Aceptada*” e informará del éxito con el mensaje “*La errata ha sido aceptada con éxito*”. En caso contrario, informará con “*Ha ocurrido un error al aceptar la errata*”. Posteriormente, con la comunicación entre ambos sistemas, el usuario móvil recibirá la leyenda actualizada.

- **Rechazar una errata**

Por otro lado, si el usuario lo considera oportuno, podrá rechazar peticiones ya que, por ejemplo, la errata está duplicada o su contenido es inadecuado. Para ello, deberá pulsar el botón correspondiente y el sistema le solicitará confirmación.

Rechazar errata

¿Está seguro que quiere rechazar esta errata?

Si

No

Figura 57. Cancelación de una errata.

Si el usuario pulsa “Sí”, el sistema marcará la errata como “Rechazada” e informará del éxito con el mensaje “La errata ha sido rechazada con éxito”. En caso contrario, informará con “Ha ocurrido un error al rechazar la errata”.

- **Detalle de una errata**

A través de esta opción, el usuario podrá consultar el historial de estados por los que ha pasado una errata. Además, podrá consultar cuál era la errata, cuál era la línea de conversación o leyenda afectada y la solución aplicada. En este caso, la errata ha sido aceptada.

Detalle de la errata

Errata

'hello' is a formal way

Oración

'Hi' or 'Hello' is an informal way to exchange greetings.

Solución

'Hi' is an informal way to exchange greetings.

Historial de estados

Fecha	Usuario	Estado
09/05/2017 16:19:32	admin	Aceptado
09/05/2017 16:09:03	appTellme	Nuevo

Cerrar

Figura 53. Detalle de una errata.

Módulo de Publicaciones

A través de este módulo, el usuario podrá consultar las publicaciones realizadas. Para ello, el usuario cuenta con varias herramientas, como un buscador de publicaciones.

Tipo

Selecciona un tipo...

Elemento

Selecciona un elemento...

Fecha publicado desde

05/05/2017

Fecha publicado hasta

Publicación

Q. BUSCAR

LIMPIAR

Figura 54. Buscador de publicaciones.

El usuario podrá buscar una publicación por los siguientes campos:

- Tipo: Identifica el tipo de publicación. Tipos:
 - Nuevo: Un elemento se publica por primera vez.
 - Actualización: Un elemento publicado ha sido actualizado.

- Elemento: Indica el elemento que ha sido publicado. Tipos:
 - Tema: El elemento publicado es un tema.
 - Conversación: El elemento publicado es una conversación.
 - Línea de conversación: El elemento publicado es una línea de conversación.
 - Unidad: El elemento publicado es una unidad.
 - Leyenda: El elemento publicado es una leyenda.
- Fecha publicado desde: Rango inferior de la fecha de publicación del elemento.
- Fecha publicado hasta: Rango superior de la fecha de publicación del elemento.
- Publicación: Descripción del elemento publicado.

Tras cumplimentar los filtros de búsqueda, el usuario deberá pulsar el botón "*Buscar*" para que el sistema recupere las publicaciones que se ajusten a lo solicitado. Para borrar los filtros establecidos, el usuario podrá pulsar el botón "*Limpiar*".

Listado de publicaciones						
Mostrar 5		Buscar por publicación ...				
Id	Publicación	Creado	Publicado	Tipo	Elemento	Usuario
14	'Hi' is an informal way ...	09/05/2017 17:28:11	09/05/2017 17:28:11	Actualización	Leyenda	admin
13	'Hi' is an informal way ...	09/05/2017 17:19:32	09/05/2017 17:19:32	Actualización	Leyenda	admin
12	Presentation	09/05/2017 17:05:08	09/05/2017 17:05:08	Nuevo	Conversación	admin
11	'Hi' or 'Hello' is an in...	09/05/2017 17:05:08	09/05/2017 17:05:08	Nuevo	Leyenda	admin
10	Greetings and Farewells	09/05/2017 17:05:08	09/05/2017 17:05:08	Nuevo	Unidad	admin

Mostrando desde 1 hasta 5 de 14 registros totales

Figura 55. Resultados de la búsqueda.

Como resultado, el usuario visualizará la siguiente información:

- Id: Número de la publicación registrada en el sistema.
- Publicación: Descripción del elemento publicado.
- Creado: Fecha de registro de la publicación.
- Publicado: Fecha de registro de la publicación ajustada a la franja horaria GMT.
- Tipo: Tipo de publicación que ha sido publicada.
- Elemento: Elemento que ha sido publicado.
- Usuario: Usuario que realizó la publicación.

Quiénes somos

En esta sección, se habla de Tecnología Social, un grupo de investigación de la Universidad de Las Palmas de Gran Canaria que cuenta con numerosos docentes e investigadores de diferentes ámbitos, y que tiene como misión el asesoramiento y el desarrollo de aplicaciones para los ámbitos social y educativo.

Por otro lado, se añade los créditos de los miembros de este Trabajo de Fin de Grado.



Figura 56. Quiénes somos

Diagrama de clases de TellMe

A continuación, se detalla las entidades utilizadas en el prototipo TellMe.

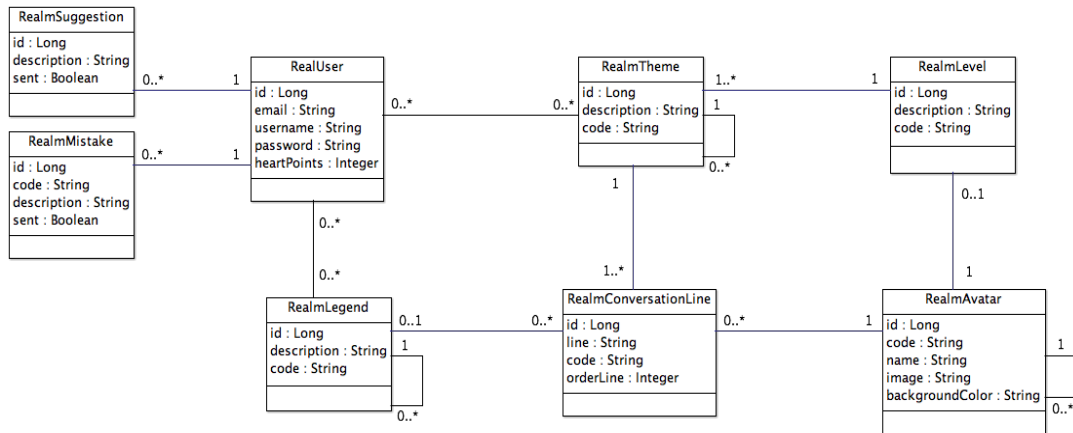


Figura 57. Diagrama de clases de TellMe.

Clase: RealmUser	
Descripción: Entidad utilizada para almacenar las cuentas de usuario.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
email String	Identifica al email del usuario y es obligatorio.
username String	Identificador del usuario. Es único y obligatorio.
password String	Clave de autenticación del usuario y es obligatorio.
heartPoint Integer	Determina el número de puntos obtenidos. Además, es requerido.
Cuando un usuario crea una cuenta, recibirá 30 puntos para comenzar a jugar.	

Clase: RealmAvatar	
Descripción: Clase que identifica los diferentes actores de la aplicación.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
code String	Código único que identifica al actor.
name String	Indica el nombre del actor. Es obligatorio.
image String	Campo requerido que identifica el icono que representa al actor.
backgroundColor String	Campo obligatorio que identifica el color asociado al actor.

Clase: RealmTheme	
Descripción: Clase que almacena los temas y las conversaciones publicadas en TellMe Server.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
code String	Código único que identifica al tema o a la conversación.
description String	Registra el título del tema o de la conversación. Es obligatorio.

Clase: RealmLegend	
Descripción: Entidad que almacena las unidades y las leyendas publicadas en TellMe Server.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
code String	Código único que identifica a la unidad o a la leyenda.
description String	Registra el título de la unidad o de la leyenda. Es obligatorio.

Clase: RealmLevel	
Descripción: Entidad que identifica los diferentes niveles de dificultad.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
code String	Código único que identifica al nivel.
description String	Registra el título del nivel. Es obligatorio.

Clase: RealmConversationLine	
Descripción: Entidad que almacena las líneas de cada diálogo.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
line String	Es obligatoria y registra la línea de la conversación.
code String	Código único que identifica la línea del diálogo.
orderLine Integer	Es obligatorio e indica la posición de la línea en el diálogo.

Clase: RealmSuggestion	
Descripción: Clase utilizada para registrar las sugerencias realizadas por el usuario.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
description String	Identifica la sugerencia realizada. Además, es obligatorio.
sent Boolean	Indica si la sugerencia ha sido enviada a TellMe Server y es requerido.
Por defecto, el atributo “ <i>sent</i> ” es falso. El proceso automático será el encargado de pasarlo a true.	

Clase: RealmMistake	
Descripción: Entidad utilizada para registrar las erratas notificadas por el usuario.	
Atributo	Descripción
id Long	Índice de la tabla en la base de datos.
code String	Identifica el código de la leyenda o de la línea de conversación que contiene la errata. Dicho código es requerido.
description String	Almacena el comentario realizado por el usuario. Además, es obligatorio.
sent Boolean	Identifica si la errata ha sido enviada a TellMe Server y es requerido.
Por defecto, el atributo “ <i>sent</i> ” es falso. El proceso automático será el encargado de pasarlo a true.	

Manual operativo de usuario de TellMe

En este manual, se ha detallado la funcionalidad que ofrece el prototipo. Además se ha completado con el diseño de la interfaz y con las explicaciones propias de un caso de uso.

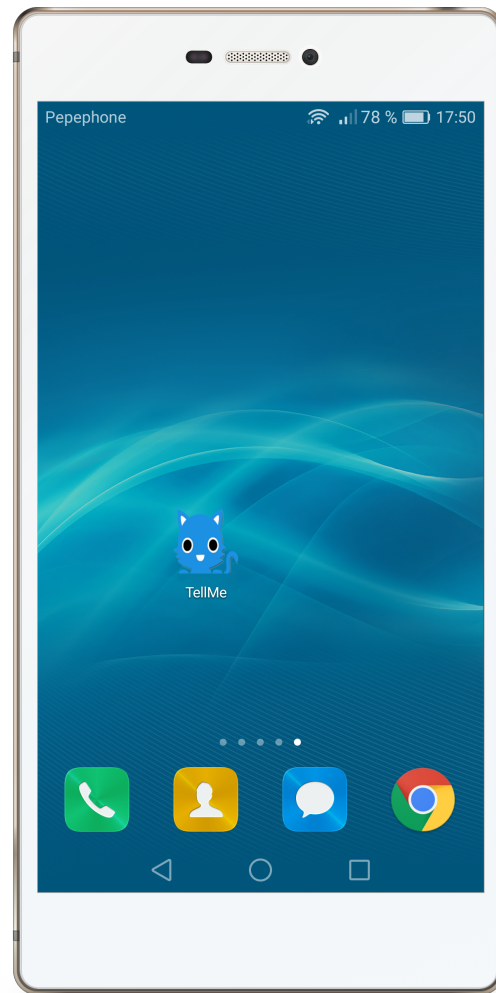


Figura 58. Acceso a TellMe

Cuando una persona instala el prototipo en su terminal, éste podrá acceder a la aplicación a través del acceso directo disponible en la pantalla principal.

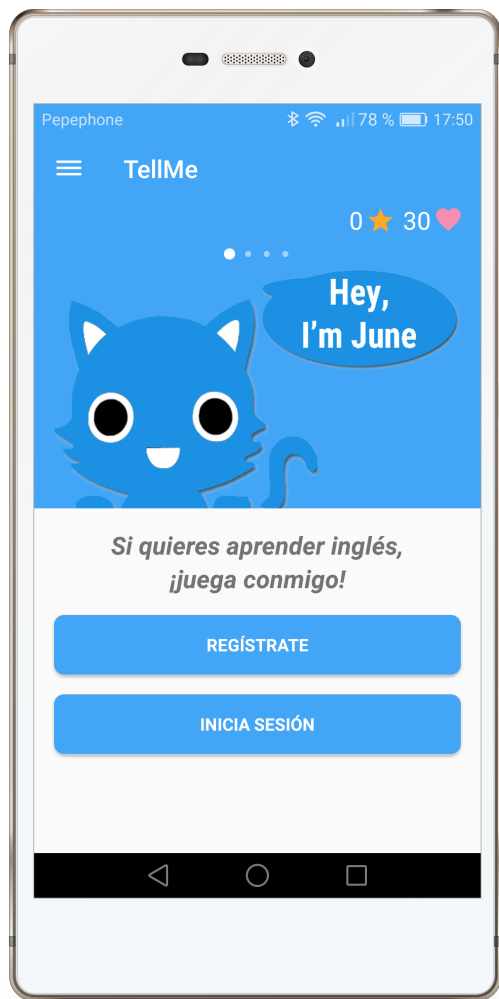


Figura 59. Pantalla principal.

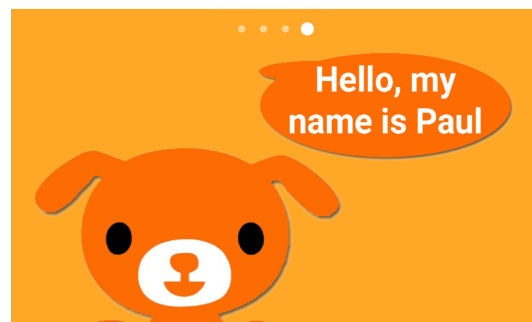
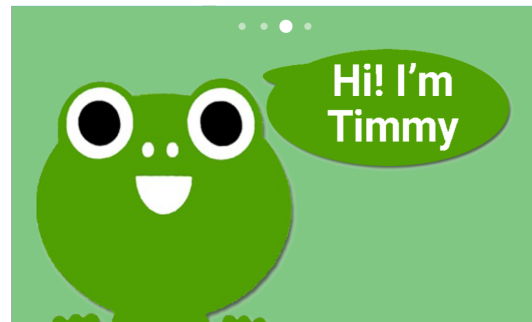
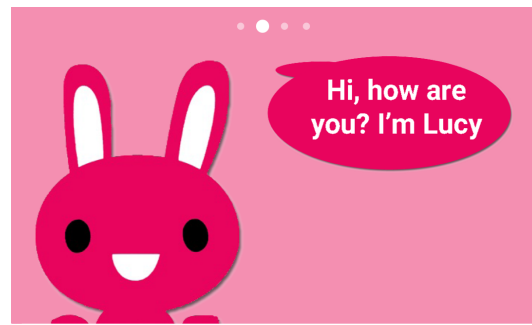


Figura 60. Presentación de actores.

En la pantalla principal, el usuario podrá conocer a June, la actriz principal del prototipo. Además, deslizando la galería de imágenes, podrá conocer al resto de actores.

Asimismo, el usuario podrá registrarse como nuevo jugador o podrá recuperar su cuenta iniciando una sesión.

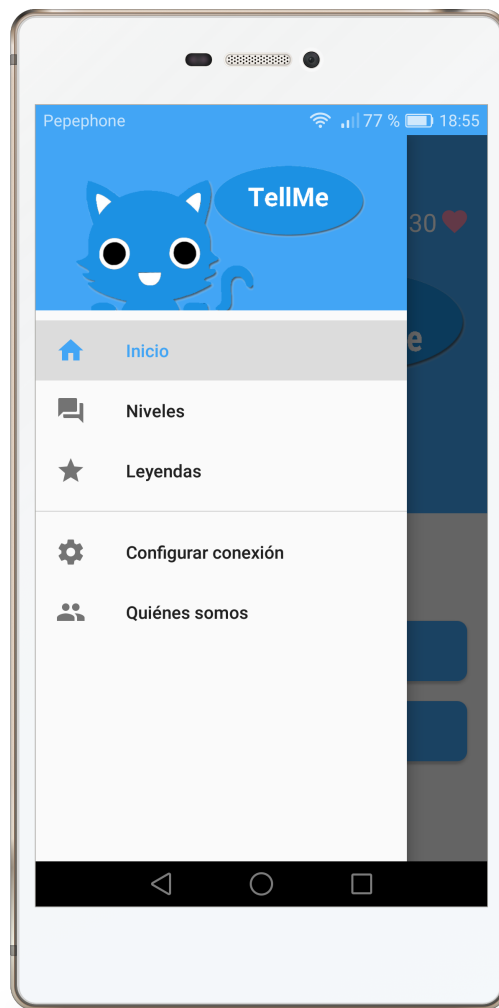


Figura 61. Menú de TellMe.

Además, el usuario cuenta con un menú donde podrá:

- **Inicio:** Regresar a la pantalla principal de la aplicación para registrarse o autenticarse.
- **Niveles:** Acceder al módulo de temas y conversaciones para jugar. Nótese que el usuario podrá jugar sin estar registrado o autenticado. Por consiguiente, el usuario parte con 30 “corazones” para jugar. Si el usuario lo estima oportuno, podrá registrarse o autenticarse y el sistema guardará su partida. En caso contrario, perderá los datos de su partida.
- **Leyendas:** Navegar a la sección de unidades y leyendas que han sido publicadas desde TellMe Server.
- **Configurar conexión:** Esta opción sólo estará disponible en la versión de desarrollo del prototipo. En él, el desarrollador podrá configurar la conexión con TellMe Server para realizar sus pruebas. En una versión candidato o producción del prototipo, nunca estará disponible esta opción, ya que la conexión estará prefijada.
- **Quiénes somos:** Esta sección habla de Tecnología Social y de los miembros de este Trabajo de Fin de Grado.

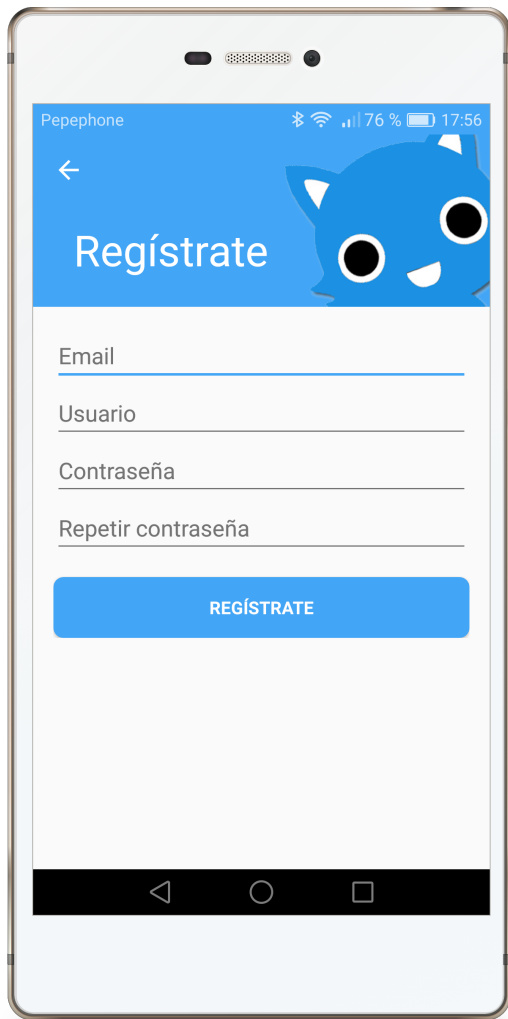


Figura 62. Pantalla de registro.

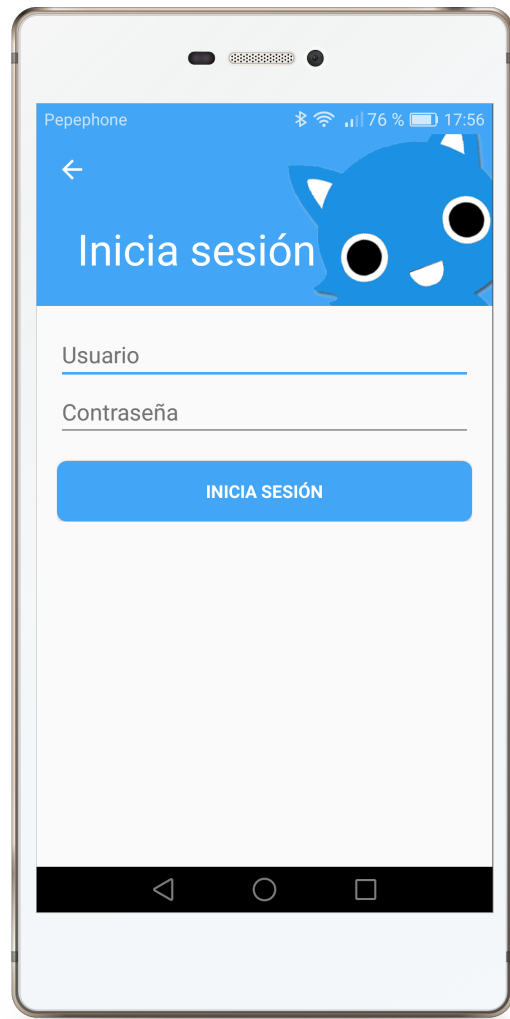


Figura 63. Pantalla de inicio de sesión .

Para crear una cuenta de usuario, el jugador deberá indicar los siguientes campos requeridos para registrarse:

- **Email:** Dato que se utilizará cuando el prototipo desarrolle la opción de recuperar contraseñas.
- **Usuario:** Deberá indicar un nombre de usuario que será único en el terminal.
- **Contraseña:** Deberá indicar una contraseña de acceso.
- **Repetir contraseña:** Por seguridad, el usuario deberá volver a escribir su contraseña.

Si el usuario posee una cuenta de usuario, podrá iniciar su sesión. Para ello, deberá indicar su nombre de usuario y contraseña. Tras la validación, el sistema cargará la partida.

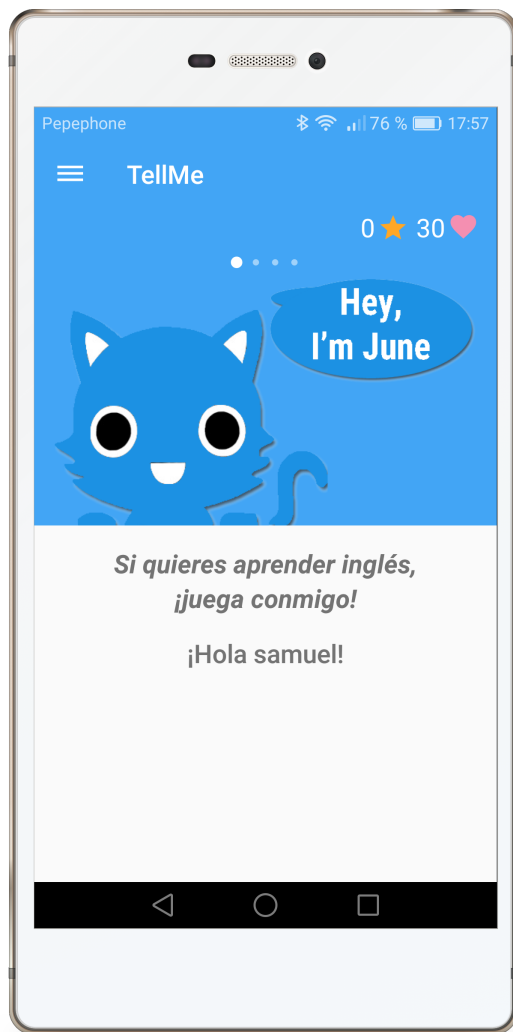


Figura 64. Usuario autenticado.

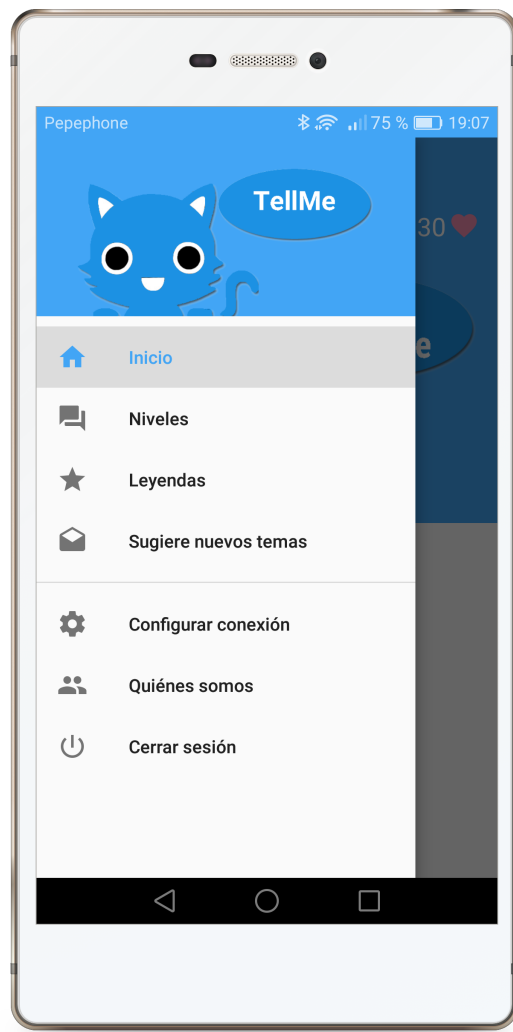


Figura 65. Más opciones tras autenticación.

Tras la autenticación del usuario, el prototipo regresa a la pantalla principal. En él, se da la bienvenida al jugador y se muestran los “corazones” y “estrellas” que haya conseguido el usuario hasta el momento.

Asimismo, se habilitarán nuevas opciones en el menú:

- **Sugiere nuevos temas:** El usuario podrá enviar sus peticiones a los administradores de TellMe Server, los cuales podrán atender su sugerencia y crear nuevos contenidos didácticos.
- **Cerrar sesión:** El usuario podrá cerrar su sesión.



Figura 66. Niveles de dificultad.

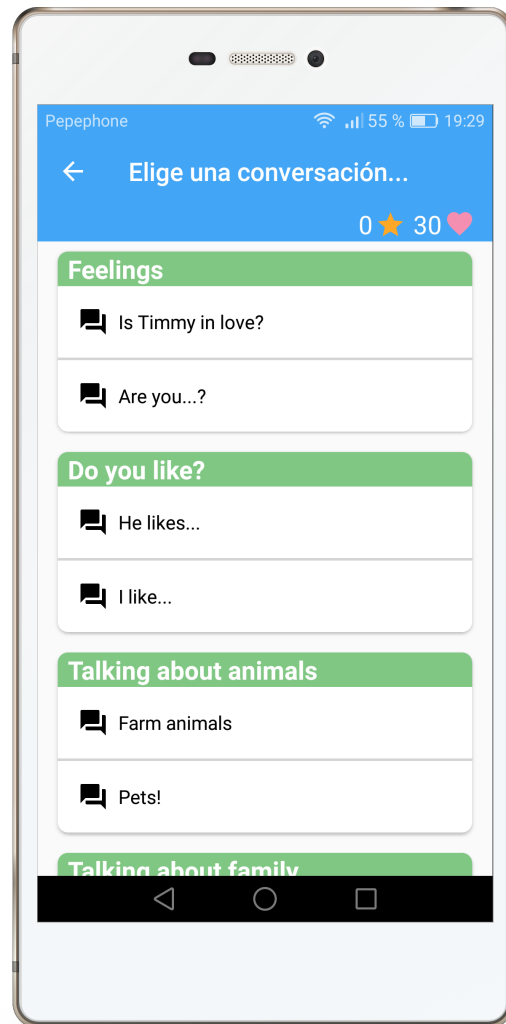


Figura 67. Temas y conversaciones.

Para comenzar a jugar, el usuario deberá pulsar la opción del menú “Niveles”. De esta manera, el sistema navegará a la pantalla correspondiente, donde el usuario deberá indicar el nivel de dificultad por el que quiere empezar (Principiante, Intermedio o Avanzado).

Posteriormente, el sistema le mostrará los diferentes temas y conversaciones que componen el nivel seleccionado. En este caso, el usuario ha seleccionado el nivel Principiante.

Por último, el usuario deberá seleccionar la conversación que más le interese y el sistema navegará al contenido de la misma.

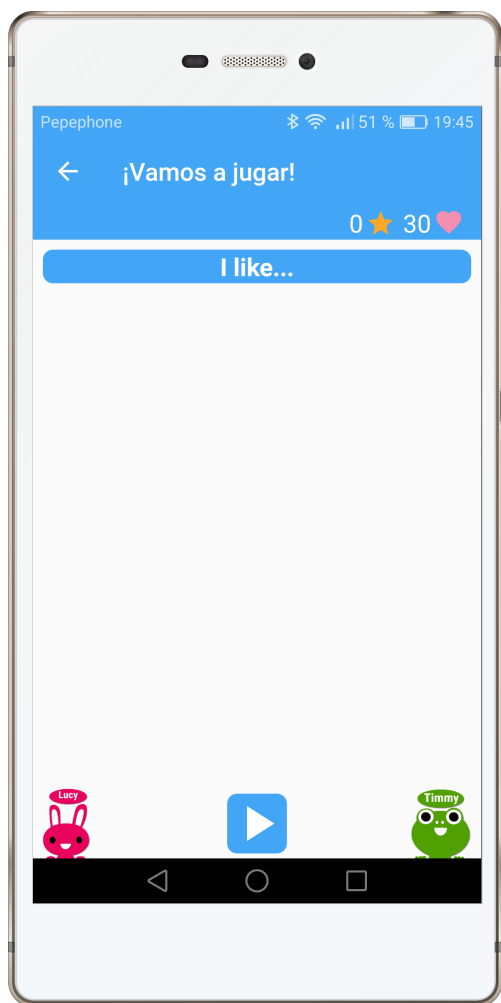


Figura 68. Pantalla de la conversación.

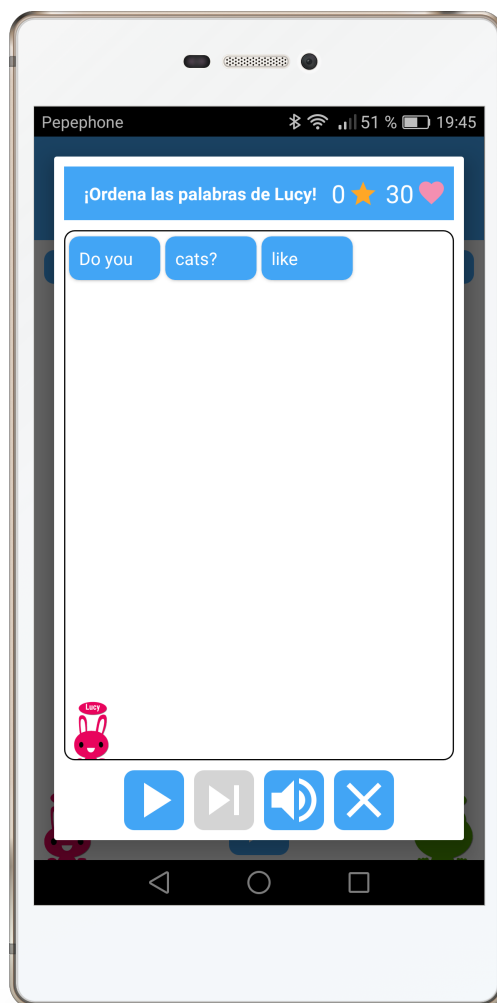


Figura 69. Construyendo líneas del diálogo.

En este caso, el usuario ha seleccionado la conversación “*I like...*”, correspondiente al tema “*Do you like?*”. En la pantalla de la conversación, el usuario podrá observar cuales son los interlocutores del diálogo seleccionado. Para iniciar, el usuario deberá pulsar el botón “*Play*”.

Como se puede observar en la Figura 69, Lucy inicia la conversación. El usuario deberá ordenar la frase del personaje y, cuando considere que la ha ordenado correctamente, deberá pulsar el botón “*Play*” para su validación.

Por otro lado, el botón del “*Audio*” siempre estará habilitado y el usuario podrá escuchar la respuesta correcta. De esta manera, reforzará su comprensión auditiva y podrá asociar el audio con el texto, lo cual, le facilitará su ordenación. Si el usuario desea abandonar la conversación, podrá pulsar el botón “*Cerrar*”.



Figura 70. Se ha descubierto una leyenda.

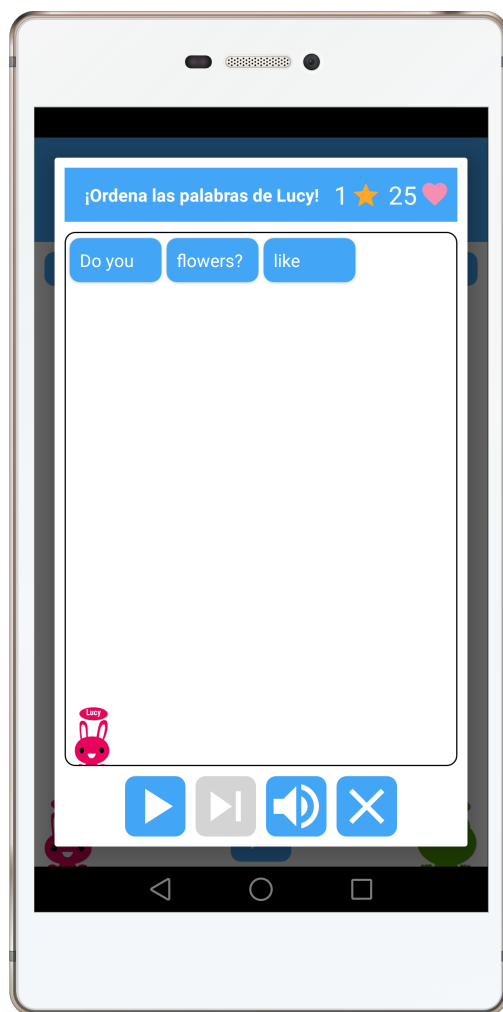


Figura 71. Otra línea de la conversación.

Tras la validación de la línea de Timmy, el usuario ha descubierto una leyenda. Este contenido didáctico extra permitirá que el usuario conozca nuevos conceptos relacionados con la temática. Por tanto, una “*estrella*” subirá al marcador del jugador.

Para continuar con el juego, el usuario deberá cerrar el diálogo y pulsar el botón “*Siguiente*”. Esto permitirá que el sistema cargue la siguiente frase. En esta ocasión, otra frase de Lucy.

La metodología del juego continúa hasta finalizar la conversación. Como se puede observar en la figura 71, el jugador tiene problemas para ordenar la frase de Lucy ya que su marcador ha bajado de 30 a 25 corazones.



Figura 72. El usuario ha gastado todos sus corazones.



Figura 73. Fin de la conversación.

En el peor caso, el usuario podrá quedarse sin puntos y el sistema le animará a volver a jugar. En caso afirmativo, el sistema reiniciará la partida con 30 corazones. Además, mantendrá las leyendas descubiertas y las conversaciones realizadas. En caso negativo, el usuario no podrá seguir jugando.

Tras terminar la ordenación de todas las líneas del diálogo, el sistema felicitará al jugador y le animará a seguir jugando. Tras finalizar cada conversación, el sistema sumará corazones al marcador del usuario.

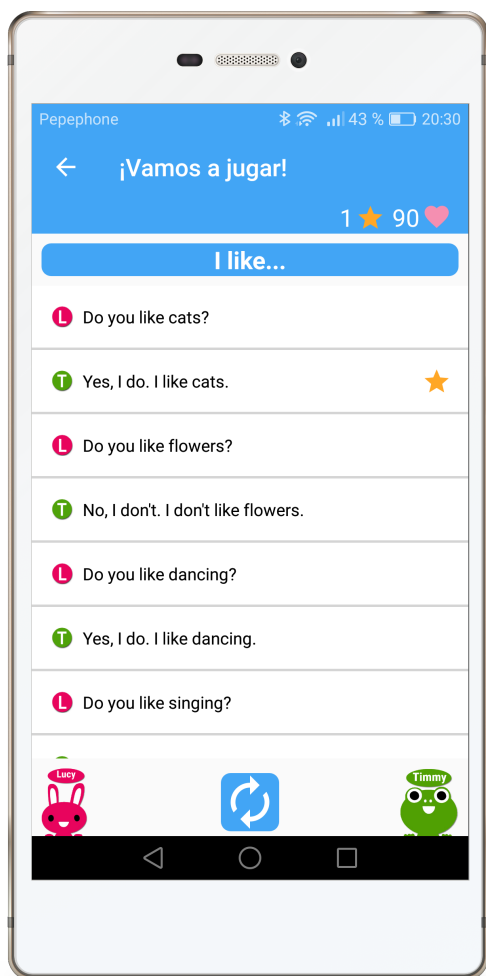


Figura 74. Conversación completada.

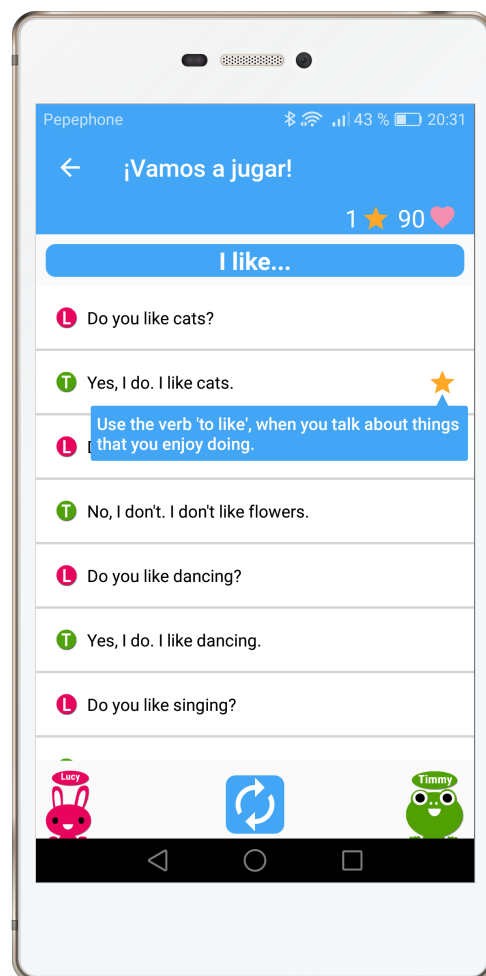


Figura 75. Consulta de la leyenda descubierta.

Tras finalizar una conversación, el usuario podrá visualizar el diálogo en su conjunto. Además, pulsando sobre cada línea, el usuario podrá reproducir el audio asociado.

Por otro lado, pulsando sobre la estrella obtenida, el usuario podrá consultar el recurso didáctico extra descubierto. Nótese que, desde la aplicación Web, el administrador puede reutilizar una leyenda en muchas conversaciones pero esto no implica que, al descubrir una leyenda previamente contabilizada, ésta suba al marcador. Sólo contabilizará una vez.

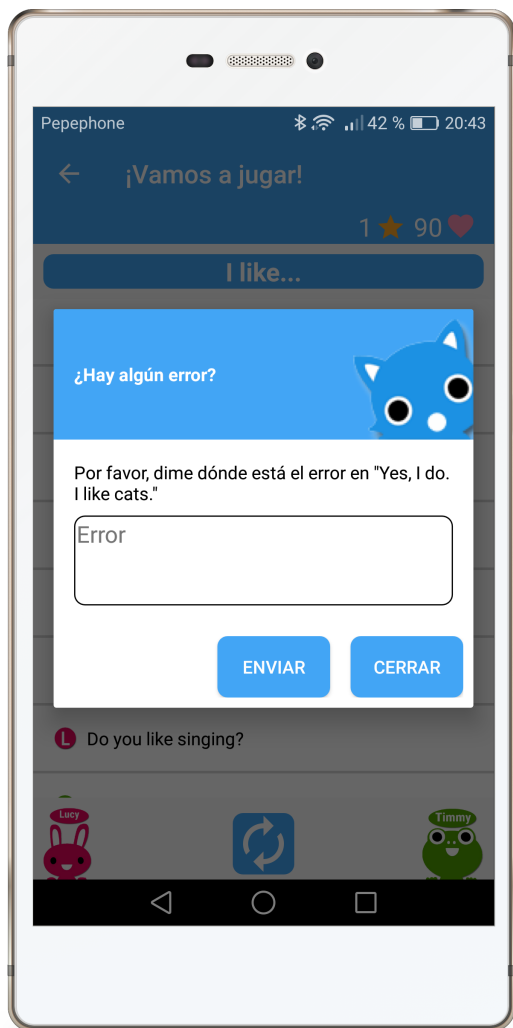


Figura 76. Notificar una errata en una línea.

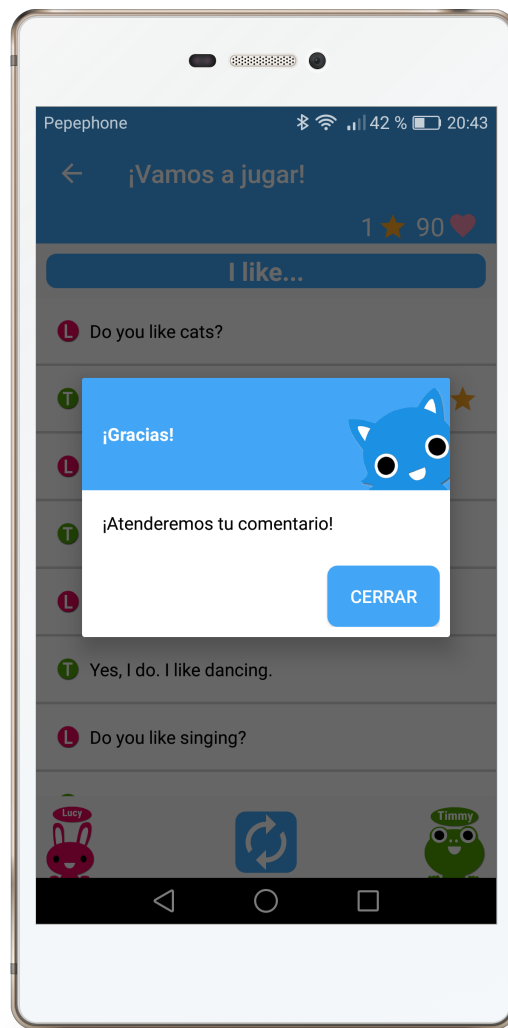


Figura 77. Enviar una errata a TellMe Server.

Por contra, si el jugador considera que la línea de algún interlocutor no es correcta, podrá pulsar sobre la inicial del interlocutor y el sistema mostrará la ventana de la figura 76. En él, el usuario podrá notificar la errata detectada. De esta manera, el administrador de la aplicación Web podrá atender la notificación y ajustar o mejorar el contenido didáctico.

Para ello, el usuario deberá pulsar el botón “*Enviar*” y el sistema almacenará la errata en su base de datos. Posteriormente, el proceso de comunicación enviará los datos al servidor Web.

Nótese que sólo se podrá notificar erratas cuando el usuario esté autenticado en el sistema. Si el usuario no se autentica, esta herramienta no estará disponible.

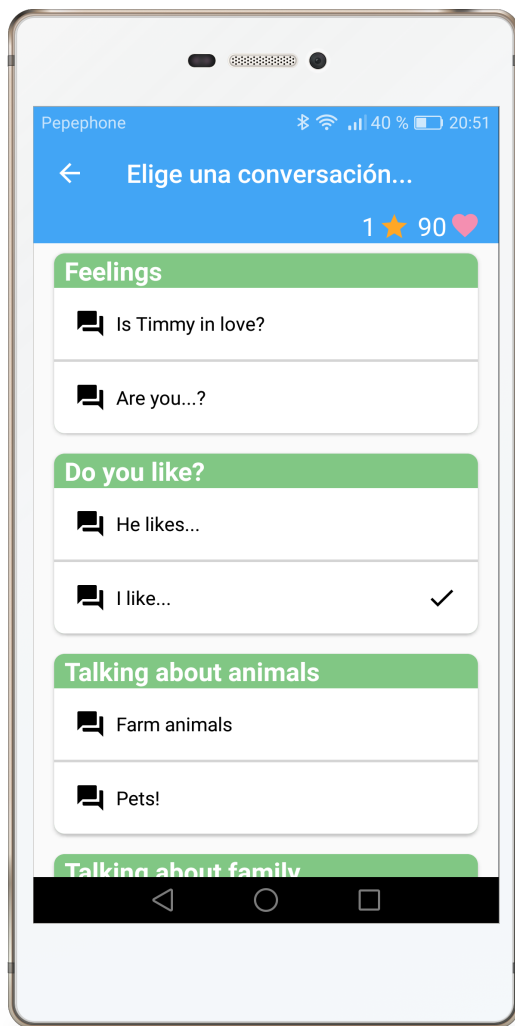


Figura 78. Temas y conversaciones.

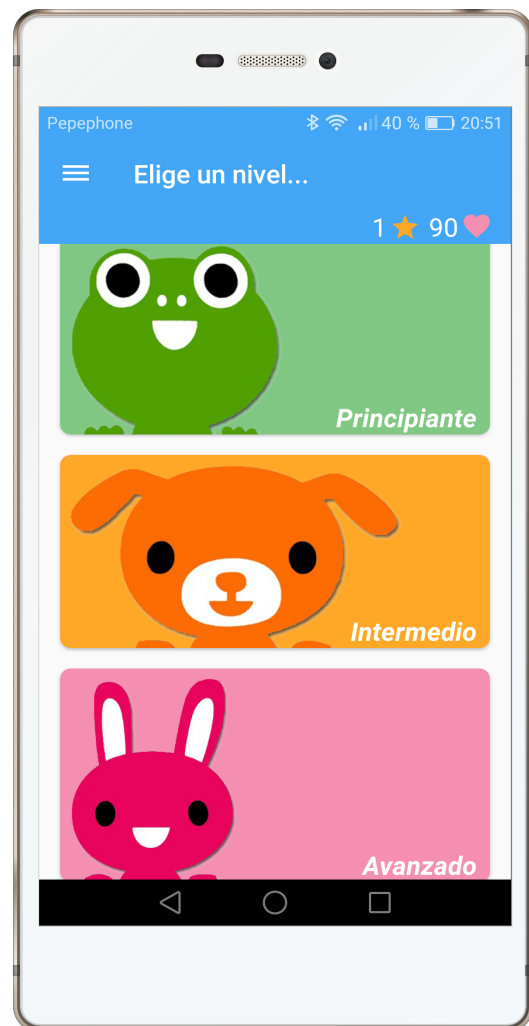


Figura 79. Niveles.

Una vez finalizada la conversación “*I like...*”, el usuario podrá retroceder en la navegación del prototipo y podrá observar que dicha conversación ha sido marcada como realizada.

El usuario podrá volver a jugar con la conversación realizada, pero el sistema no le dará más estrellas ni corazones tras volver a finalizarla. En cambio, si en el transcurso del juego, se equivoca con la ordenación de alguna frase, perderá puntos.

Finalmente, el usuario podrá seguir retrocediendo en la aplicación y llegar a la pantalla de “*Niveles*” en el cual, podrá cambiar de nivel y enfrentarse a conversaciones más complejas.

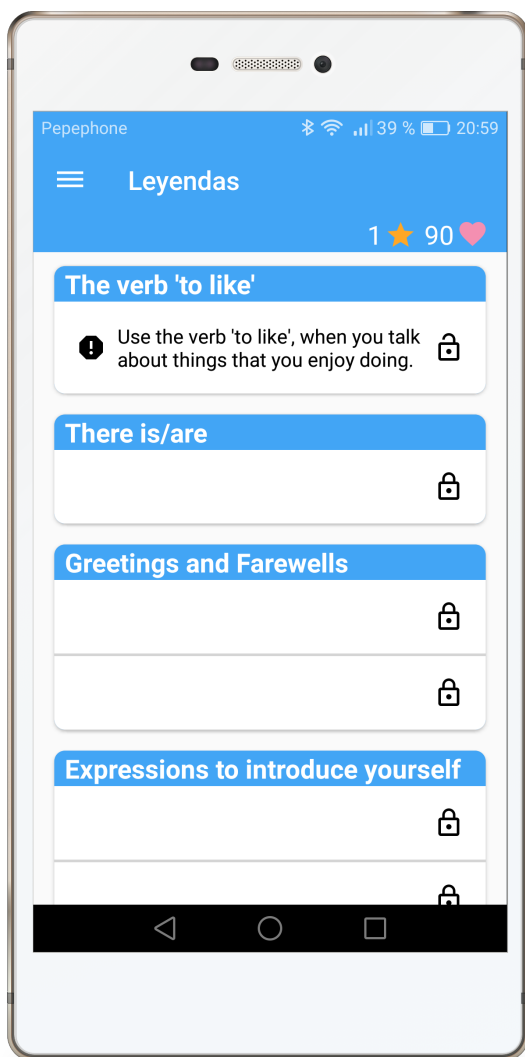


Figura 80. Sección de unidades y leyendas.

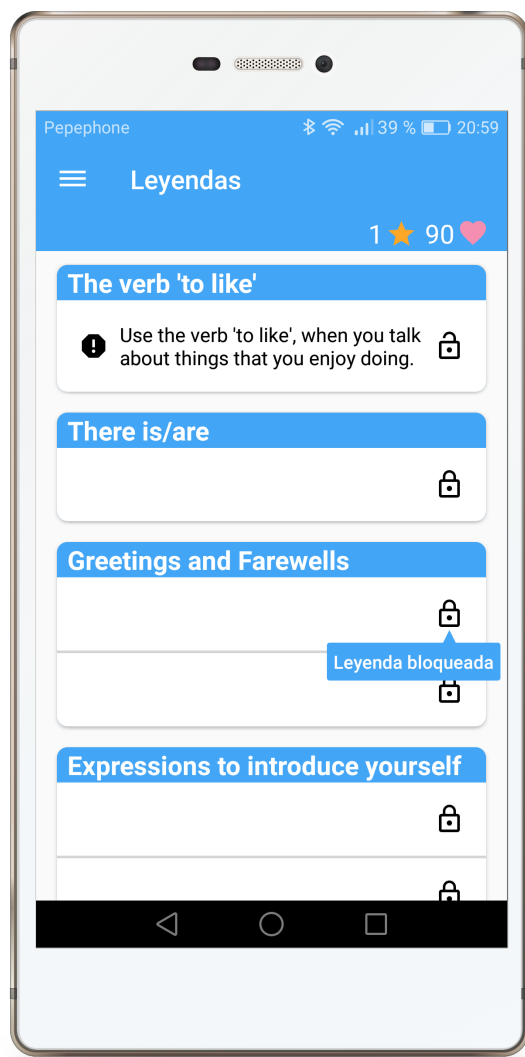


Figura 81. Leyenda bloqueada y desbloqueada.

Regresando a la pantalla principal del prototipo, el usuario podrá consultar las leyendas que ha descubierto. Para ello, deberá pulsar la opción del menú “*Leyendas*” y el sistema navegará a la sección correspondiente, mostrándole todas las unidades y leyendas que han sido publicadas desde la aplicación Web.

Por otro lado, si el usuario pulsa sobre una leyenda descubierta, podrá escucharla. De esta manera, podrá ejercitar su comprensión auditiva. Asimismo, podrá observar cuántas leyendas le quedan por descubrir.

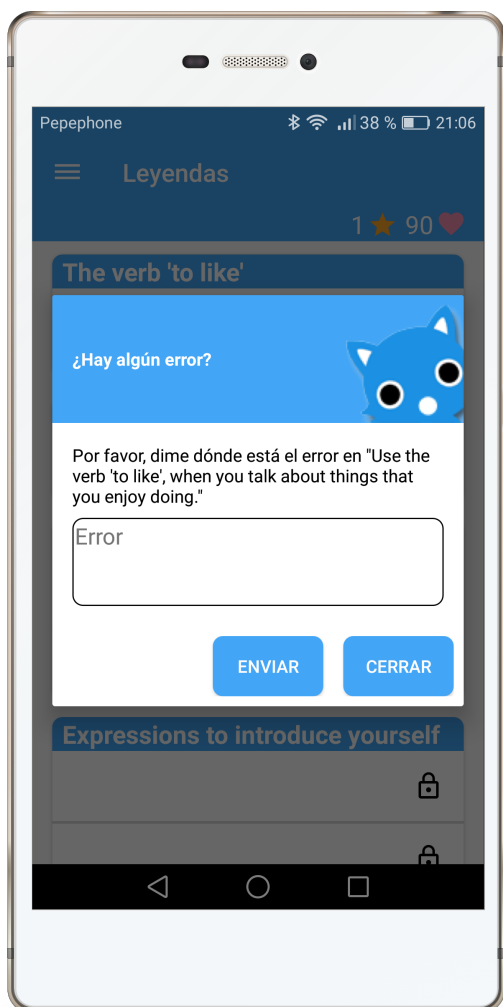


Figura 82. Notificar una errata en una leyenda.

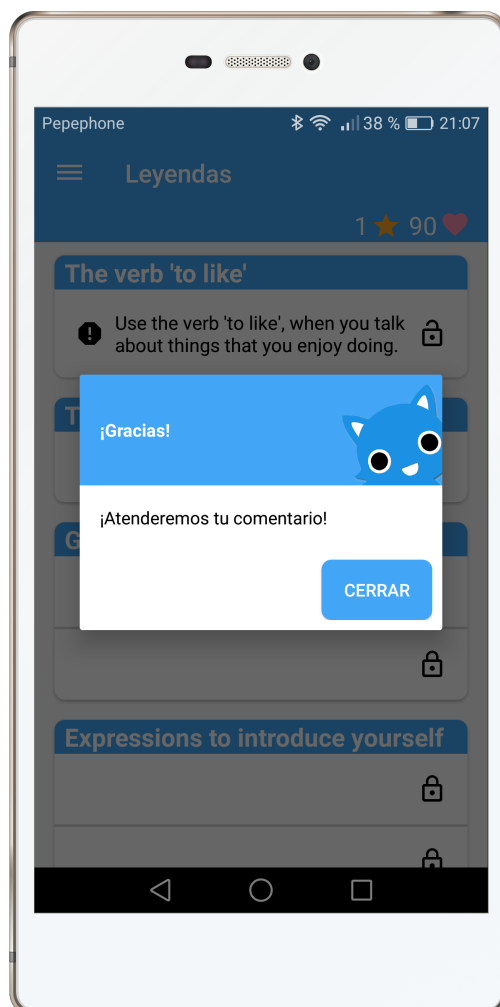


Figura 83. Enviar una errata a TellMe Server.

Por contra, si el jugador considera que una leyenda no es correcta, podrá pulsar el icono de exclamación y el sistema mostrará la ventana de la figura 82. En ella, el usuario podrá notificar la errata detectada. De esta manera, el administrador de la aplicación Web podrá atender la notificación y ajustar o mejorar el contenido didáctico.

Para ello, el usuario deberá pulsar el botón “*Enviar*” y el sistema almacenará la errata en su base de datos. Posteriormente, el proceso de comunicación enviará los datos al servidor Web.

Nótese que sólo se podrá notificar erratas cuando el usuario esté autenticado en el sistema. Si el usuario no se autentica, esta herramienta no estará disponible.

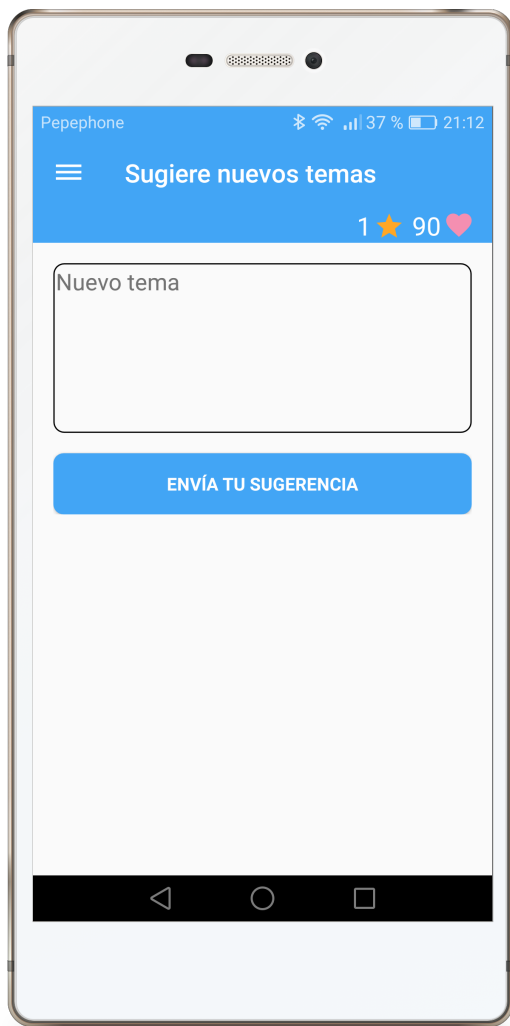


Figura 84. Sugerir un tema.



Figura 85. Enviar la sugerencia a TellMe Server.

Si el usuario lo considera oportuno, podrá sugerir nuevos temas o ideas para nuevos contenidos. Para ello, deberá pulsar la opción del menú “*Sugerir nuevos temas*” y el sistema navegará a la sección correspondiente.

De esta manera, el usuario podrá redactar su petición, ya sea en inglés o en su lengua materna, y enviarla. El sistema almacenará la sugerencia en su base de datos y, posteriormente, el proceso de comunicación la enviará al servidor Web. De esta manera, el administrador de la aplicación Web podrá atender la petición y añadir nuevos contenidos didácticos.

Nótese que sólo se podrá sugerir nuevos temas cuando el usuario esté autenticado en el sistema. Si el usuario no se autentica, esta herramienta no estará disponible.

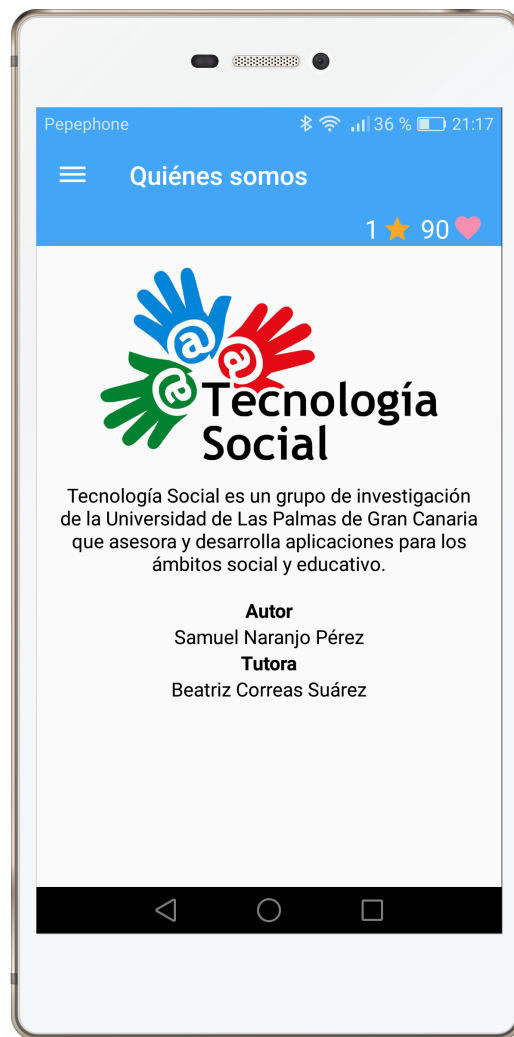


Figura 86. Quiénes somos

En esta sección, se habla de Tecnología Social, un grupo de investigación de la Universidad de Las Palmas de Gran Canaria que cuenta con numerosos docentes e investigadores de diferentes ámbitos y tiene como misión el asesoramiento y el desarrollo de aplicaciones para los ámbitos social y educativo.

Por otro lado, se añade los créditos de los miembros de este Trabajo Fin de Grado.



Figura 87. Conexión con un servidor de pruebas.

Finalmente, la opción del menú “*Configurar conexión*”, sólo se habilitará para un entorno de desarrollo. De esta manera, el desarrollador podrá configurar la conexión con TellMe Server para realizar sus pruebas.

Para ello, se deberá indicar la IP y el puerto de conexión con el servidor. Por ejemplo:

Host: `http://192.168.0.16`

Puerto: 9000

Tanto si la conexión ha sido correcta como si no, el sistema informará al desarrollador.

En una versión candidato o producción del prototipo, nunca estará disponible esta opción, ya que la conexión estará prefijada.



Figura 88. Mensaje de conexión correcta.



Figura 89. Mensaje de conexión incorrecta.

Proceso de comunicación entre sistemas

Al abrir el prototipo, éste configura varios procesos automáticos que son transparentes al usuario. Estos procesos serán los encargados de la comunicación.

```
public interface Service {  
  
    @GET("/external/publications/{id}")  
    void getNewPublications(@Path("id") Long id, Callback<NewPublication> cb);  
  
    @GET("/external/updates/{id}")  
    void getUpdates(@Path("id") Long id, Callback<UpdatePublication> cb);  
  
    @POST("/external/suggestion")  
    void postSuggestion(@Body SendSuggestion sendSuggestion, Callback<SendSuggestion> cb);  
  
    @POST("/external/mistake")  
    void postMistake(@Body SendMistake sendMistake, Callback<SendMistake> cb);  
}
```

Figura 90. Métodos utilizados en la comunicación entre sistemas.

Como el prototipo consume los datos de la aplicación Web a través de las peticiones GET, es necesario que los procesos automáticos se ejecuten periódicamente. Asimismo, si el usuario ha registrado alguna sugerencia o errata, éstas deben ser enviados al servidor a través de peticiones POST. Para ello, se ha utilizado la clase “*PendingIntent*” la cual sirve para delegar funcionalidad a otra aplicación. En este caso, los procesos automáticos serán lanzados por el sistema de alarmas del terminal y será éste quien ejecute trozos del código del prototipo.

```
public class NewPublicationsJob {  
  
    private static final int PENDING_NEW_PUBLICATION_SERVICE_ID = 15496;  
  
    public static void start(Context context) {  
        AlarmManager am = (AlarmManager)context.getSystemService(Context.ALARM_SERVICE);  
        Intent newPublicationsIntentService=new Intent(context,NewPublicationsService.class);  
  
        PendingIntent newPublicationsPendingIntent = PendingIntent.getService(context,  
                                                                                PENDING_NEW_PUBLICATION_SERVICE_ID,  
                                                                                newPublicationsIntentService,  
                                                                                PendingIntent.FLAG_NO_CREATE);  
  
        if(newPublicationsPendingIntent != null){  
            am.cancel(newPublicationsPendingIntent);  
        }  
  
        PendingIntent p_newPublicationsIntentService = PendingIntent.getService(context,  
                                                                                PENDING_NEW_PUBLICATION_SERVICE_ID,  
                                                                                newPublicationsIntentService,  
                                                                                PendingIntent.FLAG_UPDATE_CURRENT);  
  
        am.setInexactRepeating(AlarmManager.RTC_WAKEUP,  
                               System.currentTimeMillis(),  
                               60000,  
                               p_newPublicationsIntentService);  
    }  
}
```

Figura 91. Ejemplo para asociar un PendingIntent al sistema de alarmas del terminal.

Con este fin, el contexto de la actividad principal del prototipo instancia la clase “*AlarmManager*”, la cual es la responsable de gestionar las alarmas del terminal. Asimismo, se genera un “*Intent*”, es decir, un objeto que sirve para invocar otros objetos. En este caso, el Intent invocará la clase “*NewPublicationService*”.

Antes de crear la alarma, se debe comprobar si existe una alarma previa con el número definido en “*PENDING_NEW_PUBLICATION_SERVICE_ID*”. Para ello, se

pregunta por el tipo “*FLAG_NO_CREATE*”. En caso afirmativo, se cancelará la alarma configurada y se volverá a configurar con el tipo “*FLAG_UPDATE_CURRENT*”.

Finalmente, la alarma se almacena en el terminal y se ejecutará, por ejemplo, cada 60000 milisegundos, es decir, cada minuto.

```
public class NewPublicationsService extends IntentService {
    public NewPublicationsService() {
        super("NewPublicationsService");
    }

    @Override
    protected void onHandleIntent(Intent intent) {
        try{
            NewPublicationsThread thread = new NewPublicationsThread(NewPublicationsService.this);
            thread.start();
        }catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

Figura 92. IntentService definido que ejecutará la alarma.

Por consiguiente, al ejecutarse la alarma, se invocará la clase “*NewPublicationService*”, la cual extiende de la clase “*IntentService*”. Un *IntentService* es un servicio de Android que facilita la creación de hilos (Threads) o tareas asíncronas (AsyncTask), lo cual permite ejecutar tareas en segundo plano. De esta manera, no se sobrecarga el hilo principal de la aplicación y no se ralentiza.

```
public class NewPublicationsThread extends Thread {
    private Context context;

    public NewPublicationsThread(Context context) {
        this.context = context;
    }

    public void run(){
        SharedPreferences preferences = context.getSharedPreferences(GlobalKeyHelper.CONTAINER,
                                                                    Context.MODE_PRIVATE);
        Long id = preferences.getLong(GlobalKeyHelper.NEW_PUBLICATION,0L);

        Service service = ServiceGenerator.createService(Service.class);
        if(service != null){
            service.getNewPublications(id, new Callback<NewPublication>() {
                @Override
                public void success(NewPublication newPublication, Response response) {
                    try {
                        if(newPublication != null){
                            //Do something
                        }
                    }catch (Exception e) {
                        e.printStackTrace();
                    }
                }

                @Override
                public void failure(RetrofitError error) {
                    error.printStackTrace();
                }
            });
        }
    }
}
```

Figura 93. Ejemplo de hilo.

En este caso, el hilo construirá una instancia de *Service* que será la responsable de establecer la conexión con el servidor de TellMe Server. Si la conexión ha sido satisfactoria, el servicio devolverá un objeto “*NewPublication*” con los datos que ha obtenido del servidor. En caso contrario, devolverá un error.

A partir de aquí, el servicio ya dispone de los nuevos contenidos didácticos que han sido publicados en TellMe Server y podrán ser almacenados en la base de datos del prototipo.

Para establecer la conexión con el servidor, el servicio instancia la clase RestAdapter que pertenece a la librería Retrofit, a la cual le se pasará el host y el nombre de usuario y contraseña para la autenticación.

```
OkHttpClient okHttpClient = new OkHttpClient();
okHttpClient.setReadTimeout(90, TimeUnit.SECONDS);
okHttpClient.setConnectTimeout(90, TimeUnit.SECONDS);
RestAdapter.Builder builder = new RestAdapter.Builder().setClient(new OkClient(okHttpClient));
builder.setEndpoint(getHostConnection());
builder.setRequestInterceptor(new RequestInterceptor() {
    @Override
    public void intercept(RequestFacade request) {
        request.addHeader("Authorization", getUserConnection());
        request.addHeader("Accept", "application/json");
    }
});
RestAdapter adapter = builder.build();
```

Tiempo máximo de conexión y lectura de los datos.

Dirección del servidor

Usuario y contraseña de acceso

Figura 94. Ejemplo de conexión.

En un periodo máximo de 90 segundos se recibirá la respuesta JSON que es parseada a un objeto Java. Si la conexión tarda más de 90 segundos, la instancia RestAdapter aborta la petición, finalizando el hilo.

```
public class SendSuggestionThread extends Thread {
    private Context context;

    public SendSuggestionThread(Context context) {
        this.context = context;
    }

    public void run(){
        DAOHelper.init(context);
        final Suggestion suggestion = DAOHelper.findSuggestionToSend();

        if(suggestion != null){
            Service service = ServiceGenerator.createService(Service.class);
            if(service != null){
                SendSuggestion toSend = new SendSuggestion(suggestion.getDescription(),
                                                            suggestion.getUser().getUsername());

                service.postSuggestion(toSend, new Callback<SendSuggestion>() {
                    @Override
                    public void success(SendSuggestion sendSuggestion, Response response) {
                        suggestion.setSent(true);
                        DAOHelper.updateSuggestion(suggestion);
                    }

                    @Override
                    public void failure(RetrofitError error) {
                        error.printStackTrace();
                    }
                });
            }
        }
    }
}
```

Ejemplo 95. Ejemplo de una petición POST.

Si existiera alguna sugerencia por enviar al servidor, el hilo construiría una instancia de Service que enviaría ésta. Si la respuesta es satisfactoria, el sistema marcará la sugerencia como enviada.

CONCLUSIONES Y TRABAJOS FUTUROS

Llegado a este punto del prototipo, el siguiente paso sería publicar TellMe en la plataforma de distribución de aplicaciones para Android (Google Play Store). Así, se podría conocer el nivel de aceptación que tendría la aplicación por parte de los usuarios. Para ello, se deben tener en cuenta las siguientes recomendaciones:

TellMe Server

- Montar la aplicación en un servidor del grupo de investigación Tecnología Social. De esta manera, los administradores de la aplicación Web podrán generar los contenidos didácticos que utilizará el prototipo.
- Dicho servidor, deberá contar con el protocolo SSL que permitirá cifrar la comunicación entre la plataforma y la aplicación móvil.
- Añadir a la aplicación los siguientes módulos:
 - Gestión de usuarios: Facilitará la creación de cuentas de usuario y la asignación de permisos.
 - Recuperación de contraseñas: Se permitirá recuperar y/o cambiar la contraseña de un usuario.
- Montar un servidor de correo SMTP para que un usuario de la aplicación Web o del prototipo pueda recuperar y/o cambiar su contraseña. Para ello, la plataforma enviaría un correo electrónico a la cuenta de correo que el usuario facilitó al registrarse.

TellMe

- Modificar la pantalla de Registro para incluir un enlace a la política de protección de datos que haya definido el grupo de investigación. Al pulsarlo, el prototipo podría navegar a una nueva pantalla y el usuario visualizaría la información.
- Modificar la pantalla de Inicio de Sesión para añadir un enlace para la recuperación de contraseña. Al pulsarlo, el prototipo enviará la información del usuario a TellMe Server y éste enviaría un correo electrónico a la cuenta de correo que el usuario facilitó al registrarse. Tras la validación, el TellMe Server enviaría la nueva contraseña al terminal.

Observaciones generales

- El grupo de investigación Tecnología Social deberá definir su política de protección de datos, así como los términos y condiciones del uso de dichos datos. Además, deben ser accesibles desde el prototipo para que el usuario pueda consultarla.
- Se debe dar de alta un fichero en la Agencia Española de Protección de Datos (AEPD), indicando la información que se obtendrá del usuario.
- Contratar a un experto en Lengua Inglesa que cree los contenidos didácticos.

- Si el prototipo tuviera una buena aceptación por parte de los usuarios en la plataforma de distribución para Android, el prototipo podría expandirse e incluir a la Segunda Lengua Extranjera que se imparte en Educación Primaria, por lo general, el Francés. Para ello, se debe decidir cómo se incluiría el nuevo idioma:
 - Utilizando la estrategia de Duolingo, donde todos los idiomas estarían disponibles en la misma aplicación. En este caso, habría que adaptar la funcionalidad existente para que permita un nuevo idioma.
 - Siguiendo la estrategia de Babel, donde cada idioma tendría su propia aplicación. En este caso, se debería montar otra aplicación de TellMe Server para que los administradores generen los contenidos didácticos de la Segunda Lengua Extranjera. Además, se publicaría una otra versión del prototipo TellMe, denominado “*Dîtes-moi*”, que se comunicaría con el nuevo servidor.
- Con el fin de ampliar el segmento de clientes de TellMe, sería interesante desarrollar el prototipo en tecnología IOS. Para ello, el grupo de Tecnología Social podría presentarlo como propuesta de Trabajo de Fin de Grado.

Si se quisiera explotar la plataforma Web, para que los colegios generaran sus propios contenidos didácticos para sus alumnos, a continuación se presenta la siguiente idea:

TellMe Server

- Definir un nuevo tipo de cuenta de usuario. Por ejemplo, “*Alumno*”.
- Crear las cuentas de usuario de tipo “*Alumno*” de los escolares de Educación Primaria. Por ejemplo, el nombre de usuario sería el CIAL (Código de Identificación del Alumno) del estudiante.
- Añadir nuevo módulo denominado “*Seguimiento*”, donde el profesor podrá llevar un seguimiento del uso de TellMe por parte de sus alumnos.

TellMe

- Inhabilitar la descarga de contenidos didácticos hasta que el usuario se autentique.
- Pantalla de registro:
 - Se deberá mostrar un listado de los colegios que hayan adquirido la plataforma.
 - El usuario seleccionará su escuela y deberá indicar el CIAL y la contraseña que le hayan proporcionado en su colegio.
 - El prototipo navegará a la plataforma de la escuela y se autenticará, dando por válido que el estudiante pertenece a dicho colegio.
 - En este momento, se descargarán los contenidos didácticos que haya publicado la institución.
- Pantalla de inicio de sesión:
 - Cada vez que un alumno inicie sesión, se habilitarán los contenidos didácticos de su escuela.

- Procesos automáticos:
 - Envío periódico de datos del alumno. Por ejemplo, última vez que accedió al prototipo, puntos conseguidos, leyendas descubiertas o conversaciones realizadas. De esta manera, el profesor podrá llevar un seguimiento del alumno en el módulo “*Seguimiento*”.

Observaciones generales

- El grupo de Tecnología Social deberá decidir si se encarga del mantenimiento de la plataforma, donde recibiría una cuota anual por el servicio prestado o será el colegio quién se encargue del mantenimiento.
- Tras contratar los servicios de la plataforma, el grupo de Tecnología Social deberá subir una nueva versión de TellMe a la plataforma de distribución de aplicaciones para Android. En él, deberá incluir al nuevo colegio, así como la dirección de su plataforma. De esta manera, cuando un usuario seleccione su colegio, TellMe conocerá contra qué plataforma debe validar el usuario.

Conclusiones

Con este Trabajo Fin de Grado, he aprendido nuevas vertientes de la informática que, junto con mis conocimientos, me han permitido dar vida a este prototipo. Esta experiencia ha contribuido positivamente en mí, lo cual me permitirá afrontar nuevos retos profesionales.

Me siento muy orgulloso del trabajo realizado y solo tengo palabras de agradecimiento para aquellas personas que me han ayudado a llegar a la meta, aportando alguna idea, opinión o crítica. Esto me ha hecho ver que estoy rodeado de personas maravillosas que me apoyan y creen en mí. Gracias.

Instalación y configuración de TellMe Server como entorno de desarrollo

Para poder instalar TellMe Server, el host debe contar con la versión 1.8 de Java y con el gestor de bases de datos de PostgreSQL. Además, se debe seguir los siguientes pasos:

1. Descomprimir el servidor de aplicaciones de Play disponible en la siguiente ruta del repositorio:

/Código fuente/Instalador Play 2.5.10 con Activator/typesafe-activator-1.3.10-minimal.zip

- a. MacOS/Unix:
 - i. Se añade Play al PATH del sistema:
`export PATH=/path/to/activator-x.x.x/bin:$PATH`
 - ii. Se cambia los permisos del script ejecutable:
`chmod u+x /path/to/activator-x.x.x/activator`
- b. Windows:
 - i. Se añade Play al PATH del sistema:
`setx PATH=%PATH%;"C:\path\to\activator-x.x.x\bin"`

2. Descomprimir el fichero “tellmeserver-1.0-SNAPSHOT.zip” disponible en la siguiente ruta del repositorio:

/Ejecutable/TellMe Server/Entorno desarrollo/tellmeserver-1.0-SNAPSHOT.zip

- a. Crear una base de datos en el gestor de bases de datos de PostgreSQL ya sea en la propia máquina o en otra.
- b. Modificar el siguiente archivo para indicar la dirección, puerto y nombre de la base de datos donde ésta se encuentre. Además, se debe indicar el nombre de usuario y la contraseña para la conexión.
/tellmeserver-1.0-SNAPSHOT/conf/db_development.conf
- c. Cambiar los permisos del script ejecutable:
`chmod +x /tellmeserver-1.0-SNAPSHOT/bin/tellmeserver`

3. Para desplegar el servidor, se debe ejecutar el siguiente comando en el terminal. Esto creará las tablas de la base de datos en lo indicado en “db_development.conf”. Además, insertará una carga con treinta conversaciones de ejemplo.

/tellmeserver-1.0-SNAPSHOT/bin/tellmeserver -Dplay.evolutions.db.default.autoApply=true

4. TellMe Server estará arrancado en `http://localhost:9000`

5. Para acceder a la aplicación, el usuario dispone de las siguientes cuentas de usuario:

Usuario	Contraseña	Observaciones
admin	ieuf,ug	Usuario de lectura y escritura.
guest	tms,2d	Usuario de lectura

6. Para interactuar con esta versión TellMe Server, el usuario se deberá instalar el APK de TellMe, disponible en la siguiente ruta del repositorio:

/Ejecutable/TellMe/Entorno desarrollo/TellMe_v101.apk

7. Cuando el usuario acceda a TellMe, deberá configurar la conexión con TellMe Server. Para ello, deberá indicar la IP y puerto de conexión. Por ejemplo:
- Host: <http://192.168.0.16>
 - Puerto: 9000
8. Los procesos automáticos definidos en TellMe **se ejecutarán cada cinco minutos.**

Acceso al entorno candidato de TellMe Server

Para evitar que los miembros del tribunal tengan que instalar un servidor de TellMe Server, se ha preparado un entorno candidato en la siguiente dirección Web:

<http://telmeservercandidate.herokuapp.com/>

Para acceder a la aplicación, el tribunal dispone de las siguientes cuentas de usuario:

Usuario	Contraseña	Observaciones
admin	ieuf,ug	Usuario de lectura y escritura.
guest	tms,2d	Usuario de lectura

Para interactuar con esta versión de TellMe Server, el tribunal se deberá instalar el APK de TellMe, disponible en la siguiente ruta del repositorio:

/Ejecutable/TellMe/Entorno candidato/TellMe_v102.apk

En esta ocasión, no será necesario configurar la conexión con el entorno candidato, ya que estará integrada en la APK. Por otro lado, los procesos automáticos definidos en TellMe **se ejecutarán cada cinco minutos.**

A continuación, se detallan las fuentes de información utilizadas en este Trabajo Fin de Grado.

TellMe

Componentes y librerías Android

- <https://github.com/inmite/android-validation-komensky>
- <https://github.com/thesurix/gesture-recycler>
- <https://github.com/UFreedom/FloatingView>
- <https://github.com/ViHtarb/Tooltip>

Manual para desarrollar aplicaciones Android

- <https://developer.android.com/develop/index.html>

Guía para el diseño de interfaces Android

- <https://material.io/devices/>
- <https://material.io/guidelines/>

Imagen de los actores del prototipo

- <https://pixabay.com/es/etiqueta-engomada-familia-hola-1049730/>

Librería Retrofit

- <http://square.github.io/retrofit/>
- <https://futurestud.io/tutorials/retrofit-getting-started-and-android-client>

Librería Realm IO

- <https://caster.io/episodes/realm-for-android-developers/>
- <https://realm.io/docs/java/latest/>

TellMe Server

Servidor de aplicaciones Play Framework

- <https://www.playframework.com/documentation/2.5.10/Home>

Peticiones REST

- <http://www.arquitecturajava.com/servicios-rest/>

Arquitectura cliente servidor

- https://www.ecured.cu/Arquitectura_Cliente_Servidor

Framework CSS Bootstrap

- <http://getbootstrap.com/>

Librería jQuery:

- <https://jquery.com/>
- Waiting modal:
 - <https://github.com/novascreeen/jquery.waiting>
- Chosen:
 - <https://harvesthq.github.io/chosen/>
- Datatables:
 - <https://datatables.net/download/packages>

- Notificaciones:
 - <http://www.jqueryscript.net/other/jQuery-Plugin-To-Create-Animated-Bootstrap-Alerts-notify.html>
- Bootstrap Datapicker:
 - <https://github.com/uxsolutions/bootstrap-datepicker>
- Iconografía de la aplicación Web
 - <http://fontawesome.io/icons/>
- Tipografía de la aplicación Web
 - <https://fonts.google.com/specimen/Roboto>

Metodología de desarrollo

Modelo Espiral

- <http://www.ojovisual.net/galofarino/modeloespiral.pdf>

Normativa y Legislación

Competencias del Grado en Ingeniería en Informática

- http://www.eii.ulpgc.es/tb_university_ex/?q=objetivos-y-competencias-del-gii

Ley Orgánica 8/2013, de 9 de diciembre, para la mejora de la calidad educativa

- https://www.boe.es/diario_boe/txt.php?id=BOE-A-2013-12886

Decreto 89/2014, de 1 de agosto, por el que se establece la ordenación y el currículo de la Educación Primaria en la Comunidad Autónoma de Canarias

- <http://www.gobiernodecanarias.org/educacion/web/primaria/informacion/contenidos/>

Real Decreto 126/2014, de 28 de febrero, por el que se establece el currículo básico de la Educación Primaria

- <https://www.boe.es/buscar/doc.php?id=BOE-A-2014-2222>

Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal

- <https://www.boe.es/buscar/doc.php?id=BOE-A-1999-23750>

Inscripción de ficheros en el Registro General de Protección de Datos

- https://www.agpd.es/portalwebAGPD/canalresponsable/inscripcion_ficheros/index-ides-idphp.php

Educación Primaria

Programación del Liceo Cervantes

- <http://www.educacion.gob.es/exterior/centros/cervantes/es/ofertaeducativa/educacionprimaria/programacioningles.pdf>

Recursos de aprendizaje

- <https://en.islcollective.com/>
- https://procomun.educalab.es/es/ode-search?query=&type=LEARNING_RESOURCE

Recursos de la editorial Oxford Universal Press

- Primer Ciclo de Primaria
 - http://www.oupe.es/es/ELT/Primary/explorers/RecursosDestacados/Programacion CCBB Explorers 1_Prim1_Castellano.doc
 - http://www.oupe.es/es/ELT/Primary/explorers/RecursosDestacados/Programacion Explorers 2_Prim_Castellano.doc
- Segundo Ciclo de Primaria
 - http://www.oupe.es/es/ELT/Primary/explorers/RecursosDestacados/Programacion CCBB Explorers 3_Prim3_Castellano.doc
 - http://www.oupe.es/es/ELT/Primary/explorers/RecursosDestacados/Programacion Explorers 4_Prim4_Castellano.doc
- Tercer Ciclo de Primaria
 - http://www.oupe.es/es/ELT/Primary/explorers/RecursosDestacados/programacion_explorers 5_Prim5_castellano.doc
 - http://www.oupe.es/es/ELT/Primary/explorers/RecursosDestacados/programacion_explorers6_castellano.doc

Bibliografía utilizada

Explorers Class Book 6

Editorial: Oxford University Press

Edición: 2011

ISBN: 9780194509411

Autores: Sarah Phillips
Paul Shipton

Explorers Activity Book 6

Editorial: Oxford University Press

Edición: 2011

ISBN: 9780194509428

Autores: Sarah Phillips
Paul Shipton

Explorers Class Book 5

Editorial: Oxford University Press

Edición: 2011

ISBN: 9780194509244

Autores: Sarah Phillips
Paul Shipton

Explorers Activity Book 5

Editorial: Oxford University Press

Edición: 2011

ISBN: 9780194509251

Autores: Sarah Phillips
Paul Shipton