



Universidad de Jaén

Escuela Politécnica Superior de Jaén

Implementación de juegos basados en Alexa orientados a la detección de deterioro cognitivo

Autor: Sergio González Del Moral

Grado: Grado en ingeniería informática

Director: Fernando Javier Martínez Santiago

Departamento del director: Departamento de informática

Fecha: 01/07/2024

Licencia CC



(Página intencionalmente en blanco)



Universidad de Jaén

Departamento de Informática

Don Fernando Javier Martínez Santiago, tutor del Trabajo Fin de Grado titulado: **Implementación juegos basados en Alexa orientados a la detección de deterioro cognitivo**, que presenta Don Sergio González Del Moral, otorga el visto bueno para su entrega y defensa en la Escuela Politécnica Superior de Jaén.

Jaén, Julio de 2024

El alumno:

El tutor:

Sergio González Del Moral

Fernando Javier Martínez Santiago

(Página intencionalmente en blanco)

Agradecimientos

Agradezco tanto a mi tutor, Fernando Javier Martínez Santiago, como a los compañeros del departamento de psicología e informática que participaron en el proyecto CODA, permitiendo obtener los avances que en este documento son expuestos. Sin su participación, no podría haberse realizado este trabajo, especialmente sin el apoyo de Fernando y Miguel Ángel García Cumberras.

También quiero agradecer a la Universidad de Jaén por proporcionarnos un entorno de trabajo, así como a los técnicos de laboratorio del A3 que nos han ayudado a solucionar los problemas técnicos que han surgido durante el desarrollo del proyecto.

FICHA DEL TRABAJO FIN DE TÍTULO	
Titulación	Grado en Ingeniería Informática
Modalidad	Trabajo Teórico/Experimental
Especialidad (solo TFG)	Sin especialidad
Mención (solo TFG)	Sin mención
Idioma	Español
Tipo	General
TFT en equipo	No
Autor/a	Sergio González Del Moral
Fecha de asignación	21/03/2023
Descripción corta	Este trabajo consiste en el desarrollo de una aplicación que haga uso de asistentes virtuales y bases de datos para facilitar la monitorización y recolección de datos de pacientes que sean propensos a sufrir enfermedades neurodegenerativas, dando un especial enfoque a las personas de edad avanzada.

NORMAS APLICADAS EN ESTE DOCUMENTO	
LOCALES	
TFT-UJA:2017	Normativa de Trabajos Fin de Grado, Fin de Máster y otros Trabajos Fin de Título de la Universidad de Jaén (Normativa marco UJA aprobada en Consejo de Gobierno)
TFT-EPSJ:2017	Normativa sobre Trabajos Fin de Grado y Fin de Máster en la Escuela Politécnica Superior de Jaén (Normativa EPSJ aprobada en Junta de Escuela)
TFT-EPSJ	Criterios de evaluación y normas de estilo para TFG y TFM de la Escuela Politécnica Superior de Jaén
NACIONALES E INTERNACIONALES	
ISO 2145:1978	Documentación - Numeración de divisiones y subdivisiones en documentos escritos
UNE 50132:1994	Traducción de la ISO 2145
APA 6ª edición	Estilo de referencias y citas de APA (American Psychological Association)

NORMAS UTILIZADAS COMO BASE O REFERENCIA	
NACIONALES	
UNE 157001:2014	Criterios generales para la elaboración formal de los documentos que constituyen un proyecto técnico
UNE 157801:2007	Criterios generales para la elaboración de proyectos de sistemas de información
<i>Estas normas se han utilizado como base o referencia para la inclusión de algunos contenidos y definiciones sobre elaboración de proyectos, entendiendo como proyecto la documentación consensuada entre una empresa y un cliente, que da lugar al perfeccionamiento de un contrato para la elaboración de una obra o la prestación de un servicio. Por consiguiente, no debe esperarse la aplicación de estas normas en cuanto a la completitud de los contenidos ni a la organización de los mismos.</i>	

Contenido

1	Especificación del trabajo	15
1.1	Introducción.....	15
1.2	Objetivos del trabajo	16
1.3	Antecedentes y estado del arte	17
1.4	Descripción de la situación de partida	17
1.5	Requisitos iniciales	18
1.6	Alcance	19
1.7	Hipótesis y restricciones	19
1.8	Estudio de alternativas y viabilidad.....	20
1.9	Descripción de la solución propuesta	22
1.10	Material y métodos.....	22
1.11	Tecnologías utilizadas	22
1.11.1	Alexa Skill Kit (ASK)	22
1.11.2	TypeDB.....	23
1.11.3	Python	23
1.11.4	Java	24
1.11.5	Servicios de Amazon.....	24
1.12	Metodología de desarrollo de software.....	24
1.13	Análisis de riesgos	25
1.14	Estimación del tamaño y esfuerzo	27
1.15	Planificación temporal	27
1.16	Presupuesto	32
1.16.1	Software	32
1.16.2	Hardware	32
1.16.3	Personal	33
1.16.4	Desglose final	33
1.17	Normas y referencias.....	34
1.17.1	Mecanismos de control de calidad	34
2	Diseño inicial	34
2.1	Especificaciones del sistema	36
2.2	Análisis y diseño del sistema	38
2.2.1	Análisis y diseño de las Skills.....	39
2.2.2	Análisis y diseño de la aplicación de Java	42
2.2.3	Análisis y diseño de las bases de datos	44
3	Desarrollo	46
3.1	Tarea 1: Aprendizaje de tecnologías	46
3.2	Tarea 2: Diseño y desarrollo de las aplicaciones	46
3.2.1	Pruebas de verificación del sistema.....	47
3.2.2	Pruebas de validación del sistema.....	47
3.3	Tarea 3: Diseño de la base de datos	47
3.3.1	Pruebas de verificación del sistema.....	47
3.3.2	Pruebas de validación del sistema.....	48
3.4	Tarea 4: Conexión de las aplicaciones con Backend	48
3.4.1	Pruebas de verificación del sistema.....	49

3.4.2	Pruebas de validación del sistema.....	49
3.5	Tarea 5: Inclusión de APL.....	49
3.5.1	Pruebas de verificación del sistema.....	50
3.5.2	Pruebas de validación del sistema.....	50
3.6	Tarea 6: Aplicación de Java.....	51
3.6.1	Pruebas de verificación del sistema.....	51
3.6.2	Pruebas de validación del sistema.....	51
3.7	Pruebas finales	52
4	Conclusiones y trabajos futuros.....	57
5	Definiciones y abreviaturas.....	57
6	Bibliografía	59

Índice de ilustraciones

Ilustración 1 Representación de la hipótesis.....	16
Ilustración 2 Metodología incremental	24
Ilustración 4 Diagrama de Gantt 1	29
Ilustración 5 Diagrama de Gantt 2	29
Ilustración 3 Diagrama de Gantt 3	30
Ilustración 6 Representación del modelo	35
Ilustración 7 Caso de uso para jugar partida	38
Ilustración 8 Caso de uso para recoger datos	38
Ilustración 9 Storyboard pantalla inicial	39
Ilustración 10 Storyboard pantalla famosos	40
Ilustración 11 Diagrama de secuencia de Famosos	41
Ilustración 12 Diagrama de secuencia de Secuencia numérica	42
Ilustración 13 Storyboard de aplicación java 1.....	43
Ilustración 14 Storyboard de aplicación java 2.....	43
Ilustración 15 Esquema de la base de datos CodaKoba.....	44
Ilustración 16 Esquema de la base de datos famosos	45
Ilustración 17 Transacción get_user_name	48
Ilustración 18 APL táctil	50
Ilustración 19 Formato del fichero CSV	51
Ilustración 20 Juego Famosos 1	52
Ilustración 21 Juego Famosos 2.....	53
Ilustración 22 Juego Famosos 3.....	53
Ilustración 23 Juego Famosos 4.....	54
Ilustración 24 Juego Secuencia numérica 1	54
Ilustración 25 Secuencia numérica 2	55
Ilustración 26 Secuencia numérica 3	55
Ilustración 27 Servidor Web	55
Ilustración 28 Pantalla de nombre - Aplicación Java.....	56
Ilustración 29 Pantalla de correo – Aplicación Java.....	56

Índice de tablas

Tabla 1.13-1 Riesgos	26
Tabla 1.16.1-1 Costes de software	32
Tabla 1.16.2-1 Costes de hardware.....	33
Tabla 1.16.3-1 Costes de personal.....	33
Tabla 1.16.4-1 Coste final	33
Tabla 2.1-1 Historia de usuario del sistema.....	37

1 ESPECIFICACIÓN DEL TRABAJO

En este capítulo se presenta la especificación del trabajo, con una estructura y contenidos **inspirados** en los criterios y recomendaciones que establece la norma UNE 157801:2007 - “*Criterios Generales para la elaboración de proyectos de Sistemas de Información*”.

A lo largo del documento se utilizarán términos y acrónimos cuya descripción aparecen en el apartado 5 (Definiciones y abreviaturas).

1.1 Introducción

Con el aumento de esperanza de vida provocado por los avances tecnológicos y científicos de las últimas décadas, la población se ha visto afectada por un desgaste físico en las personas de edad avanzada. Según los datos del Ministerio de política territorial y función pública de 2018, el número de personas mayores de 64 años tuvo una variación positiva desde el 2001 hasta el 2018 en un 26,5% [1].

Este suceso tiene por consecuencia un aumento de la vulnerabilidad de la población al padecimiento de enfermedades que pueden impedir el desarrollo de una vida normal, como por ejemplo las enfermedades neurodegenerativas como el Alzheimer. Por este motivo, la medicina y la psicología específicamente han puesto un gran esfuerzo para estudiar y comprender las causas de estas enfermedades, cómo tratarlas y los síntomas que pueden indicar que una persona padece de estas.

Para mejorar la calidad de vida de las personas, es de especial importancia tener los medios necesarios para hacer un diagnóstico temprano de estas dolencias, con objetivo de poder tratarlas en fases iniciales, ralentizando su empeoramiento y permitiendo al paciente llevar una vida normal. Por este motivo, es esencial hacer un seguimiento de las capacidades cognitivas de las personas de edad avanzada.

El proceso de seguimiento y monitorización del mayor implica a su vez una recogida de información, que debe ser almacenada para su posterior análisis. Al ser grandes volúmenes de datos, esto implica la necesidad de usar Tecnologías de la

Información y la Comunicación (TIC), en forma de bases de datos centralizadas, para dar soporte a este procedimiento. Además, para la recolección de datos, es posible usar otras tecnologías, como los asistentes virtuales, que permiten automatizar el proceso y monitorizar a grandes cantidades de pacientes sin necesidad de intervención humana, lo cual aligera la carga de trabajo de los sanitarios, así como darle un trato y experiencia a la persona mucho más personalizada. Por último, esta tecnología tiene el añadido de poder usarse de manera completamente remota, permitiendo al mayor usarla sin necesidad de desplazarse a un centro de salud.

1.2 Objetivos del trabajo

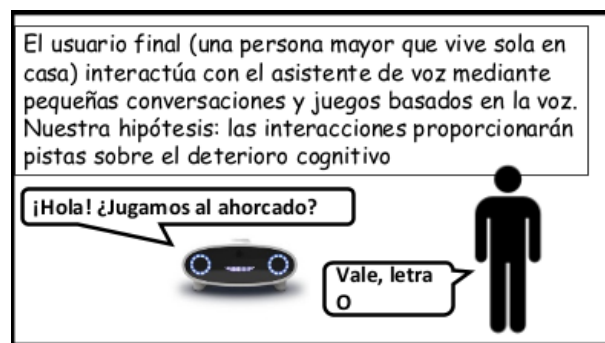


Ilustración 1 Representación de la hipótesis

El proyecto se basa en la siguiente hipótesis: es posible encontrar una correlación entre el desempeño en minijuegos y la memoria operativa. Por lo tanto, el objetivo principal de este proyecto es el desarrollo de un prototipo para una herramienta de monitorización y recogida de datos mediante el uso de minijuegos que permita al psicólogo llevar un seguimiento de la capacidad de memoria de trabajo de sus pacientes. Los objetivos específicos del proyecto son:

- Una aplicación en un asistente virtual, en este caso Alexa, accesible desde cualquier lugar, que permitirá recoger los datos.
- Un sistema de almacenamiento de datos para la su centralización y conservación, que permita un fácil acceso a estos para su posterior análisis.
- Una aplicación que permita al especialista acceder a estos datos.

1.3 Antecedentes y estado del arte

Debido a la amplia utilidad de los asistentes virtuales basados en diálogo, han habido grandes avances en su desarrollo, sobre todo por parte de grandes compañías como Google [2], Amazon [3], Microsoft [4] o Samsung [5]. En los últimos años, estas compañías han enfocado una gran parte de sus esfuerzos en el desarrollo de estas herramientas para apoyar sus tecnologías y brindar al usuario una experiencia más guiada y fácil en el uso de estas. Viendo el éxito de estos desarrollos, se pretende usar esta tecnología como puente entre el usuario (el potencial enfermo) y el diagnóstico y tratamiento de enfermedades mentales, eliminando las connotaciones negativas que tienen en muchas culturas la visita a profesionales psiquiatras.

Acompañando a los asistentes de voz, pretendemos hacer uso de Serious games que permitan al mayor entrenar su memoria y entretenerse mediante diferentes retos, facilitando la recogida de la información y haciendo ameno el uso de sistema para el paciente.

En el campo de la psicología, ha habido varios estudios [6][7][8][9][10] sobre las causas del Alzheimer y posibles marcadores para su diagnóstico, tanto de carácter biológico como psicológico. Algunos estudios importantes sobre estos posibles marcadores muestran avances alentadores para el diagnóstico precoz de esta enfermedad. Con este proyecto, se espera poder hacer uso de algunos de estos marcadores, así como encontrar más.

También cabe destacar que no es la primera vez que se usan estas tecnologías en el estudio del Alzheimer u otras enfermedades mentales, aunque en su mayoría su uso ha sido planteado para tratar las enfermedades una vez diagnosticadas, dejando el estudio y diagnóstico a los profesionales del campo.

1.4 Descripción de la situación de partida

Gracias a los recursos proporcionados por la universidad, disponemos de equipos de sobremesa en los que realizar el trabajo, así como de un servidor del Centro de Estudios Avanzados en TIC (CEATIC) en el que alojar los recursos necesarios. También se ha proporcionado para el desarrollo del proyecto una serie de dispositivos de Alexa, concretamente 2 echo show 10, que se nos ha permitido usar para el desarrollo y pruebas del software. También se puso a nuestra disposición

durante un periodo del proyecto un servidor alojado en Amazon, aunque posteriormente se descartó por motivos que se detallarán en siguientes apartados.

También se nos propocionó un laboratorio en el A3 en el que poder trabajar, así como un laboratorio domotizado, simulando una vivienda, en el C6, donde poder realizar las pruebas con mayores.

1.5 Requisitos iniciales

A continuación, se exponen los requisitos de los entregables del proyecto a alto nivel:

- **Aplicación de asistente virtual:**
 - Se debe poder interactuar con la aplicación desde cualquier lugar, siempre que se disponga de un dispositivo.
 - Se debe asegurar que el uso de la aplicación sea sencillo y claro, con motivo de facilitarle a la persona mayor la interacción.
 - Se debería poder usar en cualquier dispositivo con acceso al asistente virtual Alexa.
 - Se debe conectar la aplicación con una base de datos de manera transparente al usuario.
- **Base de datos:**
 - Se deben almacenar los datos que recopile la herramienta para su posterior análisis.
 - Se debe asegurar la integridad y continuidad de los datos
- **Aplicación para acceso a la base de datos:**
 - Se debe facilitar el acceso a los datos de la base de datos sin necesidad de conocer los esquemas que sigue.
 - Se debería poder filtrar los datos que se quieren consultar.

1.6 Alcance

- **Aplicación de asistente virtual:** El entregable consistirá en una o varias aplicaciones desarrolladas para el asistente virtual Alexa, en lenguaje Python, que permitan a los mayores interactuar con el aparato mediante juegos sencillos o conversaciones que estimulen su memoria, a fin de poder llevar un seguimiento de su posible deterioro. Los programas, alojados en AWS, se conectarán con la base de datos y almacenarán todas las respuestas del usuario para su posterior análisis. También consultará información del paciente de la base de datos para personalizar la experiencia de usuario, y si el mayor deja de usar la aplicación por un tiempo, se le recordará mediante algún tipo de notificación.
- **Base de datos:** El entregable consistirá en una base de datos de TypeDB que se conectará con la aplicación mediante su librería en Python para almacenar toda la información necesaria para el funcionamiento de los juegos y la recopilada de los pacientes.
- **Aplicación de acceso a la base de datos:** El entregable consistirá en una aplicación en Java que permita al psicólogo obtener los datos recopilados de los mayores. Se debería poder filtrar la información por usuario, de manera que se pueda estudiar cada caso separado.

1.7 Hipótesis y restricciones

El TFG se define como una asignatura de 12 créditos, lo que supone que la duración total del proyecto sería de 300 horas, incluyendo todas las etapas del ciclo de vida, con la excepción del mantenimiento. Sin embargo, al ser parte de un proyecto de investigación llevado a cabo por la universidad, se trabajó durante un año completo en él, exceptuando el mes de agosto.

También debemos tener en cuenta las limitaciones que Alexa supone para el proyecto. En primera instancia, y como es lógico, al ser un programa de terceros, no tenemos acceso a su código, y cambios en su política de uso o funcionamiento pueden suponer retrasos importantes en nuestro proyecto. Siguiendo con las restricciones que nos impone, esta herramienta impone restricciones en el tiempo de respuesta de un usuario, que debe ser menor o igual a 6 segundos, extensible hasta 8. Esto supone

limitar la complejidad de los juegos para asegurar que los mayores tengan tiempo suficiente para pensar la respuesta y responder.

Por último, hay que destacar que, pese a la disposición de 2 echo show 10 para el desarrollo del proyecto, es posible que se necesite la adquisición de otros modelos para realizar pruebas con ellos, a fin de asegurar que las aplicaciones funcionen correctamente.

1.8 Estudio de alternativas y viabilidad

Previo a la realización del proyecto, se han estudiado diferentes alternativas respecto a las tecnologías principales del proyecto: Los asistentes virtuales y las bases de datos.

Para los asistentes virtuales, se buscó principalmente que cumpliera las características de disponibilidad, facilidad de uso y programabilidad. Ahora, enumeraré las diferentes alternativas estudiadas y sus aspectos destacables:

- **Asistente virtual de Google:** El asistente virtual de Google tiene la ventaja de estar disponible de manera gratuita en casi cualquier tipo de dispositivo. Cualquier teléfono Android o iPhone dispone de acceso mediante su App Store, así como los altavoces de Google, la pantalla inteligente Nest Hub, televisiones de diferentes vendedores, como LG, NVIDIA o Sony, y los wearables de Samsung. Por otro lado, su facilidad de uso es otro gran punto a favor, ya que la frase de activación, “Hey Google”, es fácil de recordar. Por desgracia, las posibilidades que ofrece para programar aplicaciones son totalmente precarias, impidiendo la realización del proyecto mediante esta alternativa.
- **Asistente de Microsoft, Cortana:** El asistente de Microsoft, Cortana, que fue retirado como aplicación independiente en la primavera de 2023, fue otra de las alternativas planteadas. Este asistente podía usarse desde cualquier dispositivo con sistema operativo Windows 10 u 11, actualmente limitándose a Windows 11, dispositivos Alexa, Microsoft 365 y Outlook. Al no ser programable, se descartó rápidamente como posible alternativa.
- **Asistente de Samsung, Bixby:** El asistente de Samsung, Bixby, es accesible mediante cualquier producto de Samsung posterior al Samsung

Galaxy S8. Tiene un alto grado de programabilidad, pudiendo hacer aplicaciones en JavaScript y su propio entorno de desarrollo. También cabe destacar la simplicidad en su uso, que es personalizable mediante entrenamiento, permitiendo al desarrollador elegir la frase de activación de su aplicación. Por desgracia, no parece que se pueda mantener una conversación con la estructura necesaria para permitir realizar juegos de memoria, con lo cual se descartó.

- **Asistente de Amazon, Alexa:** El asistente de Amazon, Alexa, está disponible mediante cualquier dispositivo Alexa y la mayoría de los dispositivos móviles, aunque en un menor grado de funcionalidad. En el caso de las Skills, los programas desarrollados para Alexa, solo están disponibles mediante sus dispositivos. Estas Skills son altamente programables, pudiendo elegir entre varios lenguajes de programación como Java, Python, C#, Go, Ruby o Powershell. Su uso es personalizable mediante entrenamiento, tal y como lo es Bixby, y permite una mayor libertad en el flujo de aplicación que el anterior. Por estos motivos, se decidió elegir esta alternativa para el desarrollo del proyecto.

En cuanto a las alternativas de bases de datos, se buscó principalmente que tuviera la capacidad de inferir datos y que fuera sencilla de acceder de manera externa. Las alternativas estudiadas son:

- **MongoDB [11]:** Una base de datos Not only Structured Query Language (NoSQL), que permite conectar con cualquier aplicación desarrollada en JavaScript, Python, Java, C++ o C# mediante su librería. Dispone de inferencia de tipos. Pese a cumplir los requisitos para el proyecto, se optó por otra alternativa, que se expondrá más adelante.
- **TypeDB [12]:** Una base de datos NoSQL tipada, capaz de conectar con aplicaciones desarrolladas en Python, Rust, Node.js, Java, C, C++ y C#. Dispone de inferencia de datos, pudiendo asignar reglas que completen los datos almacenados de los usuarios mediante información ya almacenada en la base de datos, así como un soporte completo a los grafos de propiedades y el modelo entidad-relación-propiedad. Se optó por esta alternativa por la facilidad de aprendizaje de su lenguaje, así como por la

recomendación del jefe de proyecto, Fernando Javier Martínez Santiago, por la oportunidad de aprendizaje sobre los grafos de propiedades que supone su uso.

1.9 Descripción de la solución propuesta

Se realizarán una serie de minijuegos de memoria operativa como Skills de Alexa, en el lenguaje de programación Python, que permitan recoger datos y almacenarlos en una base de datos TypeDB hospedada en el servidor del CEATIC, a la que se podrá acceder mediante una aplicación de java que permita recoger la información almacenada de manera simple para el psicólogo. Ya que los propios juegos necesitarán hacer uso de recursos multimedia, estos se hospedarán en el servidor del CEATIC mediante un servidor web con certificado Secure Socket Layer (SSL), requerido por Alexa para ser accesible desde una Skill [13]. Este servidor web dará acceso a una estructura de directorios donde se almacenarán los archivos MPEG-2 Audio Layer III (MP3) e imágenes necesarias para los minijuegos.

1.10 Material y métodos

Para el seguimiento del trabajo y las versiones, se ha usado un repositorio de Git, y para documentos compartidos se ha usado Google Drive. En cuanto al software usado para desarrollar las skills, se usó la plataforma web de Amazon, Alexa Developer Console, donde se permite la creación, prueba y puesta en marcha de las Skills. Por último, para desarrollar el programa de Java, se usó Apache Netbeans, un entorno de desarrollo integrado (IDE) usado principalmente para Java.

1.11 Tecnologías utilizadas

En este apartado expondré las tecnologías utilizadas en este proyecto, así como la justificación para su uso.

1.11.1 Alexa Skill Kit (ASK)

ASK es un conjunto de herramientas que Amazon proporciona a los desarrolladores para implementar aplicaciones, llamadas Skills, para Alexa. Dispone

de varios lenguajes de programación, anteriormente mencionados en las alternativas estudiadas, y permite un gran control sobre el funcionamiento del programa.

Se puede desarrollar en un entorno local o usar el entorno web que proporciona Amazon, que facilita el desarrollo de la Skill y su validación. En este proyecto, se ha optado por usar el entorno web. Para gestionar el contenido multimedia, se ha usado Alexa Presentation Language (APL), que es un formato estructurado derivado de JavaScript Object Notation (JSON) que define lo que se muestra por la pantalla del dispositivo.

En cuanto a su funcionamiento, hay que comentar varios aspectos relevantes de Alexa. Como la mayoría de los algoritmos de generación de texto, usa un algoritmo de aprendizaje automático que genera texto usando la palabra más probable. Sin embargo, al ser una skill que hemos desarrollado, en lugar de usar esa generación de texto, usa el guión que nosotros le pasamos. Para analizar lo que dice el usuario, pasa la interacción de voz a texto y busca patrones que se correspondan con el contenido de los slots configurados o los intents existentes, siguiendo la lógica de dialogo especificada en la Skill que se este usando en ese momento.

1.11.2 TypeDB

TypeDB es una base de datos NoSQL que usa un esquema para definir la estructura de la información a guardar. Usa el modelo Entidad-Atributo-Relación y es capaz de crear nuevas entidades, atributos o relaciones que hereden propiedades de otras creadas anteriormente. También se pueden definir reglas que permiten a la base de datos inferir información a través de los datos almacenados. Por último, su lenguaje de consultas es fácil de usar y entender, debido a que imita el lenguaje natural.

1.11.3 Python

Python es el lenguaje de programación principal que se ha usado en el proyecto. Se eligió principalmente por la facilidad de uso y su eficiencia, así como la familiaridad que poseo al haber trabajado previamente con este. Las Skills de Python y el código que conecta las Skills con la base de datos está desarrollado con este lenguaje, usando la correspondiente librería de TypeDB.

1.11.4 Java

Java es el segundo lenguaje de programación que se ha usado en el proyecto. Debido a que TypeDB solo puede usarse a través de su lenguaje de consulta, se diseñó una aplicación en Java para permitir a los psicólogos acceder a los datos de los pacientes sin necesidad de aprender a manejar la base de datos.

1.11.5 Servicios de Amazon

Durante el desarrollo, como ya se expuso anteriormente, se tuvo acceso a un servidor S3 de Amazon Web Services, en el que se hospedaron los contenidos multimedia necesarios para el proyecto. En un principio se eligió esta solución por su compatibilidad con ASK, que facilitaba el uso de los archivos multimedia en la Skill sin tener que pasar por un servidor web, pero finalmente, debido al precio y algunos problemas en su configuración, se decidió cambiar al servidor del CEATIC.

1.12 Metodología de desarrollo de software

En la consecución del proyecto, se ha utilizado una metodología de trabajo incremental, que nos permite realizar cambios en la aplicación tras recibir el feedback de las pruebas hechas por los psicólogos. Se consideró adecuado debido a la flexibilidad necesaria en el desarrollo de los juegos, que estuvieron sujetos a varios cambios para adaptarse a la necesidad de los psicólogos. Esto se produjo mediante reuniones regulares con el equipo de psicología para concretar cambios o características necesarias de los minijuegos y los datos almacenados.

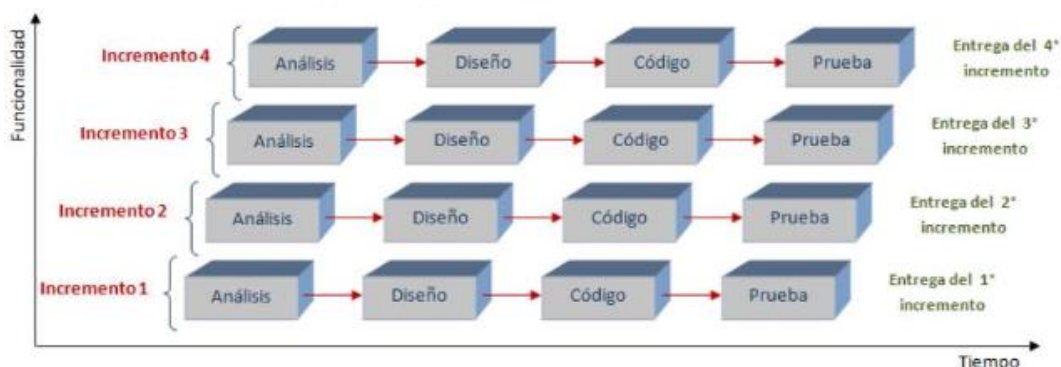


Ilustración 2 Metodología incremental

1.13 Análisis de riesgos

En este apartado, se presenta un análisis de los riesgos del proyecto, concretando su impacto, probabilidad, a qué activos afecta y qué medidas se tomaron para lidiar con ellos. En la tabla al final del apartado se usará una escala de 1 a 10 para representar la probabilidad y el impacto del riesgo.

- **Perdida de datos:** La pérdida de datos de los pacientes era uno de los principales riesgos a tener en cuenta durante la consecución del proyecto. Durante el mismo, hubo dos ocasiones en la que la base de datos fue completamente borrada, una durante el cambio de S3 al servidor del CEATIC y otra durante una caída del servidor. Para mitigarlo, se realizaban copias de seguridad cada semana, exportando la base de datos de TypeDB. Estas se guardaban tanto en el equipo local del laboratorio como en el Drive del proyecto. La pérdida de esta información hubiera supuesto, dependiendo del momento, entre varias semanas a dos meses de trabajo para los psicólogos, que tendrían que realizar las pruebas con los pacientes de nuevo.
- **Problemas legales con la recopilación de datos:** Al ser datos de carácter clínico y personal, hay que informar a los usuarios sobre su recopilación. Por suerte, las propias Skills de Alexa pueden configurarse para pedir permiso para la recopilación de datos para su uso. Para evitar, además, recopilar información si se lanzara la Skill oficialmente, para las pruebas se hacían en un entorno cerrado mediante un beta test, funcionalidad que la propia developer console de Alexa también ofrece. De esta manera, controlamos qué información recopilamos, asegurándonos de tener permiso de los usuarios. Por último, para la recogida de datos del proyecto se pidió permiso al comité de ética de la Universidad de Jaén.
- **Avería del material hardware:** Durante la consecución del proyecto, el uso repetido del hardware, especialmente los altavoces Alexa, que fueron utilizados por diferentes personas y los cuales necesitaban se tocados debido a la pantalla táctil, suponía la posibilidad de avería, ya fuera por un mal uso o un accidente durante las pruebas. Para prevenir la pausa de las

pruebas, como ya se ha dicho, se proporcionaron 2 altavoces, teniendo uno de repuesto en caso de avería.

- **Fallo externo de software:** Este riesgo es complejo de prevenir, ya que implica a terceros sobre los que no tenemos control. En nuestro caso, este riesgo afectaría sin en algún momento un cambio de Alexa o TypeDB hiciera que las skills o la base de datos dejara de funcionar, o el propio servicio de Amazon tuviera problemas. Por suerte, Amazon, al ser una gran compañía, da la seguridad de que, pese a que pueda haber un problema puntual, será solucionado con la mayor brevedad posible. En cuanto a TypeDB, para evitar este tipo de situaciones, se descargó una versión concreta del servidor y se usó durante toda la consecución del proyecto, evitando así cambios que pueda sufrir en ese periodo de tiempo.
- **Baja por enfermedad/motivos personales:** En caso de que en algún momento alguien del personal del proyecto enfermara o no pudiera realizar su trabajo por motivos personales, dependiendo de la responsabilidad, tendría que asignarse más tiempo al proyecto o sustituirla.

Riesgo	Probabilidad	Impacto	Activos	Contingencia
Pérdida de datos	6	10	Base de datos	Copias de seguridad periódicas
Problemas legales con la recopilación de datos	7	8	Base de datos Skills de Alexa	Permisos de la aplicación Control de acceso
Avería del material hardware	5	6	Altavoces Alexa Ordenadores	Hardware de repuesto
Fallo externo de software	4	7	Skills de Alexa Base de datos	Uso de una versión fija del software
Baja por enfermedad/motivos personales	6	7	Personal del proyecto	Aplazar la consecución del proyecto Reemplazar personal

Tabla 1.13-1 Riesgos

1.14 Estimación del tamaño y esfuerzo

Ya que el presente proyecto es un TFG, no existen restricciones de tipo económico, sino de tipo temporal (un número aproximado de horas). Por consiguiente, los cálculos de tamaño del proyecto están supeditados el tiempo disponible. En este caso, al ser un proyecto de investigación, se han trabajado más horas en el proyecto de lo que los créditos reflejan. En cuanto al esfuerzo, se dispone de tan solo un efectivo (la persona autora del trabajo).

1.15 Planificación temporal

Debido al modelo incremental usado en el desarrollo del proyecto, podemos dividir su desarrollo en diferentes etapas o incrementos, que vendrán expuestos en las siguientes ilustraciones y serán detalladas más adelante.

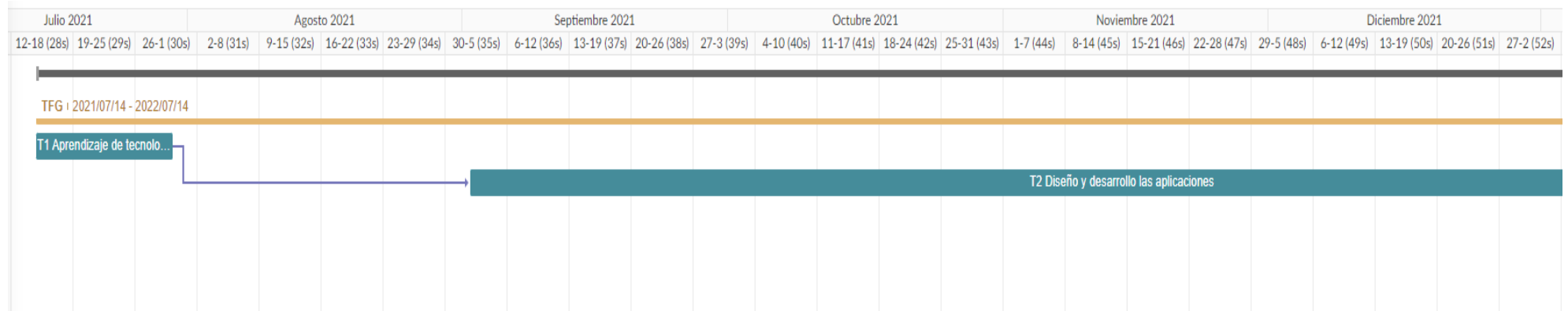


Ilustración 3 Diagrama de Gantt 1

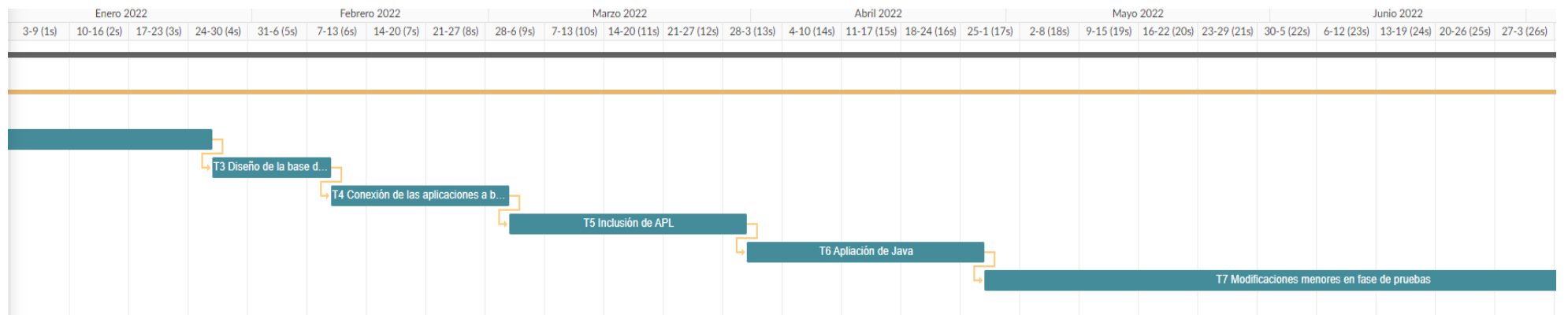


Ilustración 4 Diagrama de Gantt 2



Ilustración 5 Diagrama de Gantt 3

- Tarea 1 – Aprendizaje de tecnologías: El objetivo de esta tarea fue familiarizarme con las tecnologías de Alexa y TypeDB, aprendiendo a manejarlas y creando aplicaciones simples para realizar pruebas sobre su funcionalidad. Duró aproximadamente 2 semanas. Tras este periodo, hubo un mes de vacaciones, en el que el proyecto no avanzó.
- Tarea 2 – Diseño y desarrollo de las aplicaciones: El objetivo de esta tarea fue realizar un diseño de los minijuegos a implementar, concretando su funcionamiento, los recursos y las pautas necesarios para las pruebas, así como su implementación. Esta etapa fue la más laboriosa del proyecto, durando aproximadamente 21 semanas. Semanalmente, había reuniones con el equipo de psicólogos para refinar los detalles de los minijuegos.
- Tarea 3 – Diseño de la base de datos: El objetivo de esta tarea fue diseñar tanto la base de datos principal, como las bases de datos necesarias para cada juego, conteniendo links a imágenes y audios, así como datos particulares de cada juego. Duró aproximadamente 2 semanas.
- Tarea 4 – Conexión de las aplicaciones con Backend: El objetivo de esta tarea fue conectar los juegos con las bases de datos, permitiendo extraer información de esta y almacenar las interacciones de los usuarios. Duró aproximadamente 3 semanas.
- Tarea 5 – Inclusión de APL: El objetivo de esta tarea fue refinar el contenido de las aplicaciones con un apartado visual, permitiendo al mayor interactuar con la pantalla táctil del dispositivo. Duró aproximadamente 4 semanas.
- Tarea 6 – Aplicación de Java: El objetivo de esta tarea fue diseñar e implementar una aplicación que permitiera a los psicólogos obtener los datos de las pruebas sin necesidad de usar una consola de comandos para acceder a la base de datos. Duró aproximadamente 4 semanas.
- Tarea 7 – Modificaciones menores en fase de pruebas: El objetivo de esta tarea fue dar soporte a las aplicaciones con cambios o solucionar defectos detectados en pruebas reales con los pacientes, usando el feedback que proporcionaron los psicólogos. Duró aproximadamente 11 semanas.

1.16 Presupuesto

En este apartado se detallará el coste económico del proyecto y su justificación. Se definirá el coste en 3 categorías diferentes:

- **Software:** Se incluyen licencias de los programas y servicios necesarios.
- **Hardware:** Se incluyen el coste de los equipos específicos del proyecto.
- **Personal:** Se incluye el presupuesto destinado a la contratación del personal de gestión (jefe de proyecto) y programación (el autor del documento).

También se hará un desglose final en el que se añadirá una estimación de los costes indirectos (instalaciones y factura de servicios), calculada como el 10% del coste total.

1.16.1 Software

Aquí se incluyen los pagos correspondientes a licencias y servicios software que se usaron para el proyecto. Pese a que todos los conceptos del presupuesto suponen una duración de 12 meses (equivalente al proyecto), el servicio de S3 y EC2 de Amazon, como ya se ha visto en anteriores apartados, fue reemplazado durante el proyecto, por lo que su desglose corresponde al precio de 8 meses de contrato. Para el cálculo de los precios se ha usado el precio listado en los sitios oficiales de los proveedores [20][21].

Concepto	Descripción	Coste
Office 365 personal	Programa para documentación	3.443'88€ (6.99€/mes)
Amazon S3	Servicio de almacenamiento en la nube	18'40€ (2'30€/mes)
Amazon EC2	Servicio de cómputo en la nube	1442'48€ (180'31€/mes)

Tabla 1.26.1-1 Costes de software

1.16.2 Hardware

Aquí se incluye el precio de los dispositivos hardware empleados durante el desarrollo del proyecto, excluyendo aquellos que no fueran específicamente

adquiridos con tal propósito. Los precios listados provienen de los sitios oficiales de los proveedores [22].

Concepto	Descripción	Coste
Alexa Echo Show 10	Dispositivo Alexa con pantalla	538€ (269€/u)

Tabla 1.36.2-1 Costes de hardware

1.16.3 Personal

Aquí se incluye el coste de contratación del personal del proyecto. Al estar contratado mediante una beca de investigación, el sueldo del programador (el autor del documento) fue fijado a 500€ mensuales. El resto de los sueldos han sido sacados del BOE [23][24].

Concepto	Descripción	Coste
Programador Junior	Coste de contratación	6.000€ (500€/mes)
Jefe de proyecto	Coste de contratación	24.108€ (2.009€/mes)
Psicólogo	Coste de contratación	22.488€ (1.874€/mes)

Tabla 1.46.3-1 Costes de personal

1.16.4 Desglose final y amortización

Aquí se desglosa los costes de cada apartado y se calcula la estimación total del presupuesto del proyecto.

Concepto	Coste
Software	4.904'76€
Hardware	538€
Personal	52.596€
Costes indirectos	5.803'87€
TOTAL	63.842'63€

Tabla 1.56.4-1 Coste final

Debido a que no se dispone de licencias de software permanente, el único concepto amortizable es el hardware. Según las tablas de amortización proporcionadas por la agencia tributaria [28], el total amortizado durante la duración del proyecto (1 año) sería de 67'25€.

1.17 Normas y referencias

A continuación, se presentan las normas, reglamentos y referencias de cualquier tipo que han sido de aplicación en la elaboración del trabajo y/o en la puesta en práctica del mismo.

1.17.1 Mecanismos de control de calidad

El aseguramiento de la calidad en la redacción del trabajo implica la verificación de la completitud (falta de omisiones), la integridad de la documentación, así como una redacción clara, concisa y entendible por todos los participantes e interesados en dicho trabajo.

Al estar el trabajo desarrollado por una sola persona y verificado por un/a tutor/a, no es necesario llevar un documento de control de edición y revisión de la documentación. De esta forma, se han utilizado mecanismos básicos para la verificación de la integridad y completitud, que incluyen un control de versiones a nivel de documento y un control sencillo de la trazabilidad de especificaciones y requisitos.

2 DISEÑO INICIAL

En este apartado se mostrará el diseño del sistema de forma general, introduciendo sus componentes y justificando su uso. En los siguientes subapartados, se profundizará en este diseño.

En primer lugar, la arquitectura del sistema propuesto consiste en un modelo cliente-servidor, en el que un servidor (CEATIC) mantendrá la información requerida y el cliente (Skills o aplicación Java) hará peticiones para añadir o leer información. Aunque en su inicio el servidor, como se ha explicado anteriormente, estuviera

hospedado en un servicio de Amazon, en esta sección se especificará únicamente la arquitectura final.

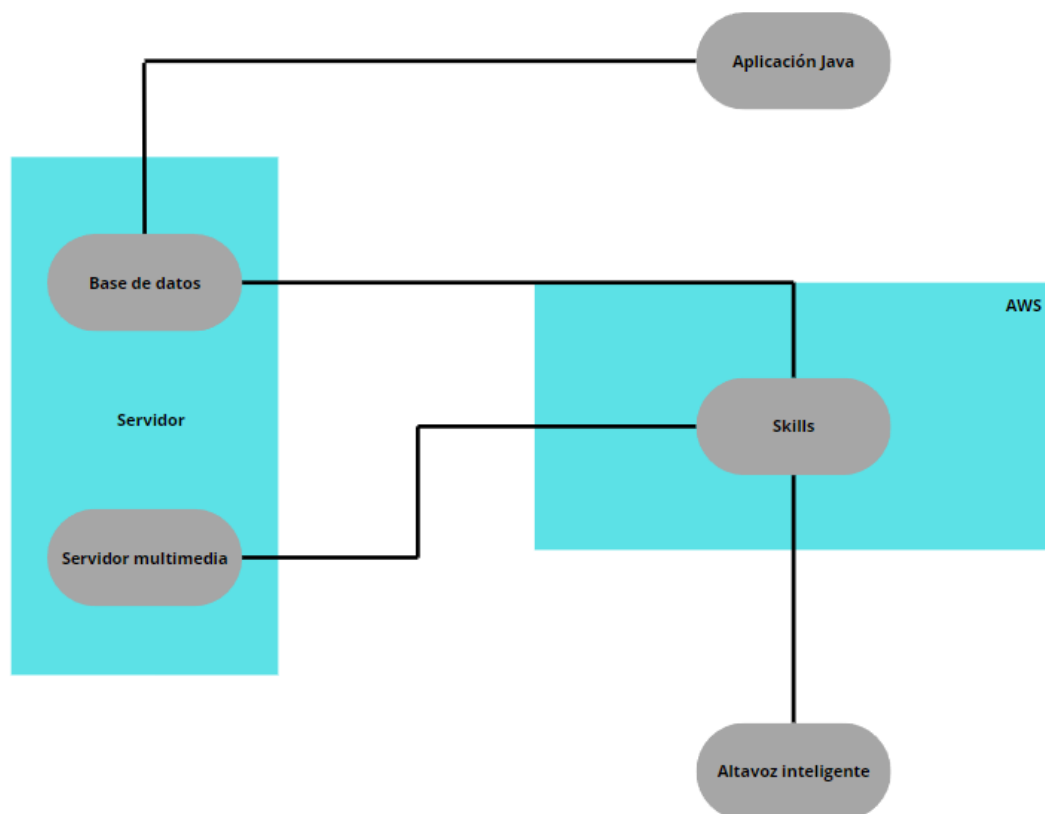


Ilustración 6 Representación del modelo

Como se puede observar en la ilustración:

- La información del juego (Skill) se recoge del servidor multimedia cada vez que inicie una nueva partida.
- La base de datos de usuarios y el servidor multimedia están alojados en el servidor del CEATIC.

- El altavoz inteligente que use el paciente solo accederá a AWS, donde está alojada la Skill de Alexa.
- La Skill será la encargada de recoger información del servidor y escribir las iteraciones del usuario en la base de datos.
- Los psicólogos tendrán acceso a la base de datos mediante la aplicación de Java, haciendo que para los usuarios el servidor sea totalmente transparente.

2.1 Especificaciones del sistema

En este apartado, se especifican los requisitos funcionales y no funcionales del sistema, así como los casos de uso. Primero, vamos a ver los requisitos funcionales:

- El paciente deberá poder iniciar un juego con el altavoz inteligente si está registrado en el sistema.
- El paciente recibirá una explicación del juego al iniciarlo.
- El paciente debe de poder dejar la partida en cualquier momento.
- El paciente puede pedir ayuda si no ha entendido las instrucciones.
- Se deben registrar todas las iteraciones del paciente con la Skill.
- La Skill debe acceder al servidor para recoger la información necesaria para el juego.
- La Skill debe dar una señal que indique el inicio del turno.
- La base de datos debe registrar interacciones del usuario de tipo texto y táctil.
- La base de datos debe contener información sobre los pacientes y las partidas de cada uno.
- El servidor multimedia contendrá los archivos de imagen y video necesarios para cada minijuego.
- El psicólogo debe poder acceder a los datos de los pacientes.

Ahora, se especificará la lista de requisitos no funcionales:

- La Skill deberá presentar el minijuego de manera clara y concisa, dando ejemplos de interacción.
- Las Skills deben funcionar en cualquier dispositivo Alexa. En caso de no funcionar por ser necesaria la pantalla, indicarlo al iniciar la Skill.
- Los recursos multimedia deben de estar lo mejor adaptados posible para personas con discapacidades sensitivas parciales.
- Los juegos deben poder adaptarse a los usuarios.
- La base de datos debe ser exportada cada semana.
- La aplicación de Java debe ser fácil de manejar para el usuario.
- La información de salida de la aplicación Java debe ser interpretable por los psicólogos.

A continuación, se exponen los diagramas de caso de uso junto a las historias de usuario:

Actor	Caso de uso	Valor aportado
Mayor	Jugar a los minijuegos con el altavoz	Ver mi progresión
Psicólogo	Obtener datos del paciente	Tener un seguimiento de las capacidades del paciente

Tabla 2.1-1 Historia de usuario del sistema

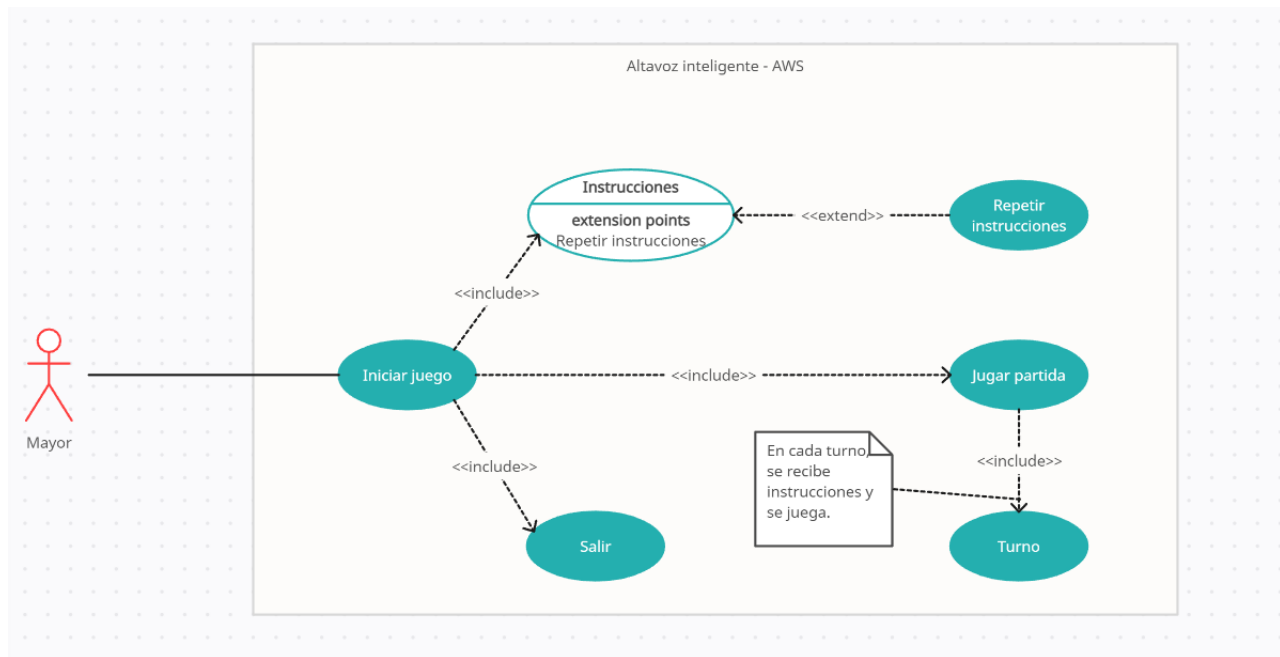


Ilustración 7 Caso de uso para jugar partida

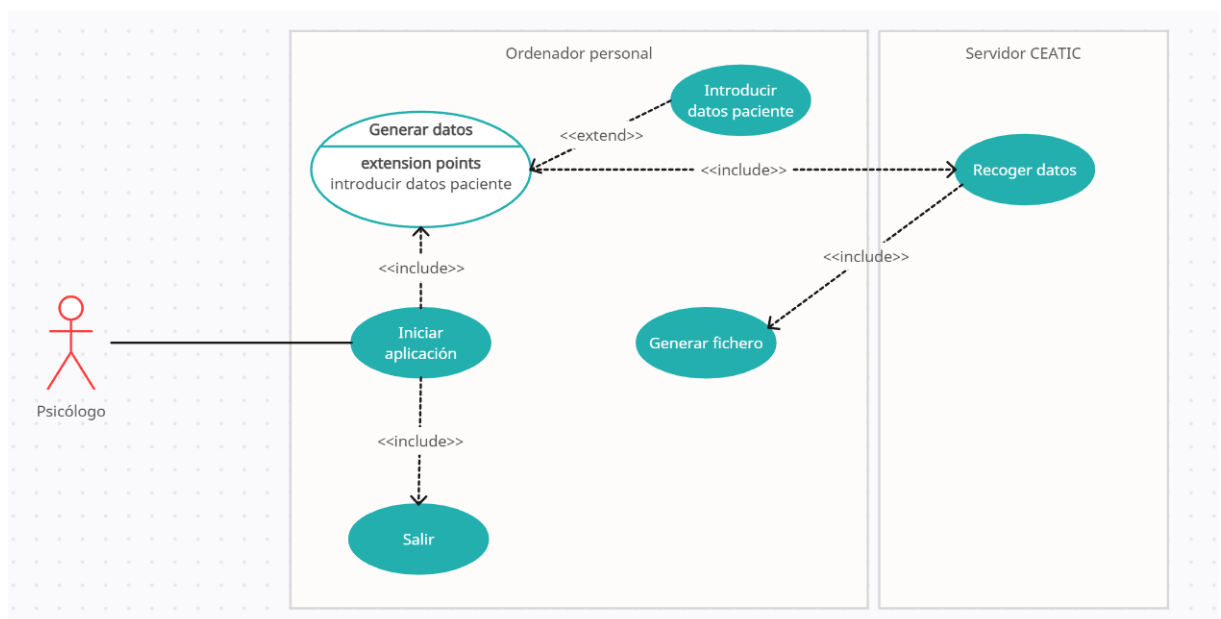


Ilustración 8 Caso de uso para recoger datos

2.2 Análisis y diseño del sistema

En este apartado se exponen el diseño del sistema, los diagramas de secuencia correspondientes a los casos de uso anteriormente detallados, el diseño de las bases de datos y el diseño de la interfaz y storyboards de las Skills.

2.2.1 Análisis y diseño de las Skills

En este subapartado nos centraremos en el diseño de la interfaz y los storyboards de las Skills y su funcionamiento básico. Cada una de las Skills tendrá una pantalla inicial, con el logo del juego centrado (En este storyboard se usa el logo del CEATIC). Arriba a la izquierda estará el nombre del juego al que se está jugando, y a la derecha aparecerá el logo del CEATIC. Al no haber herramientas de modelado gratuitas que incluyan dimensiones para diseñar aplicaciones de Alexa, es posible que las dimensiones difieran de la realidad. El diseño de la pantalla principal se muestra en la siguiente ilustración.



Ilustración 9 Storyboard pantalla inicial

En el caso del juego de famosos, que se explicará más adelante, en el apartado de desarrollo, hay una pantalla extra, que consiste en las imágenes de 3 famosos, que actúan a su vez como botones. Aunque parezca contraintuitivo poner imágenes como botones, en este caso al ser una aplicación cuyas instrucciones se escuchan al empezar la partida, y después en cada turno se explica que hacer, se considera que debe quedar claro la función de estos elementos al usuario.

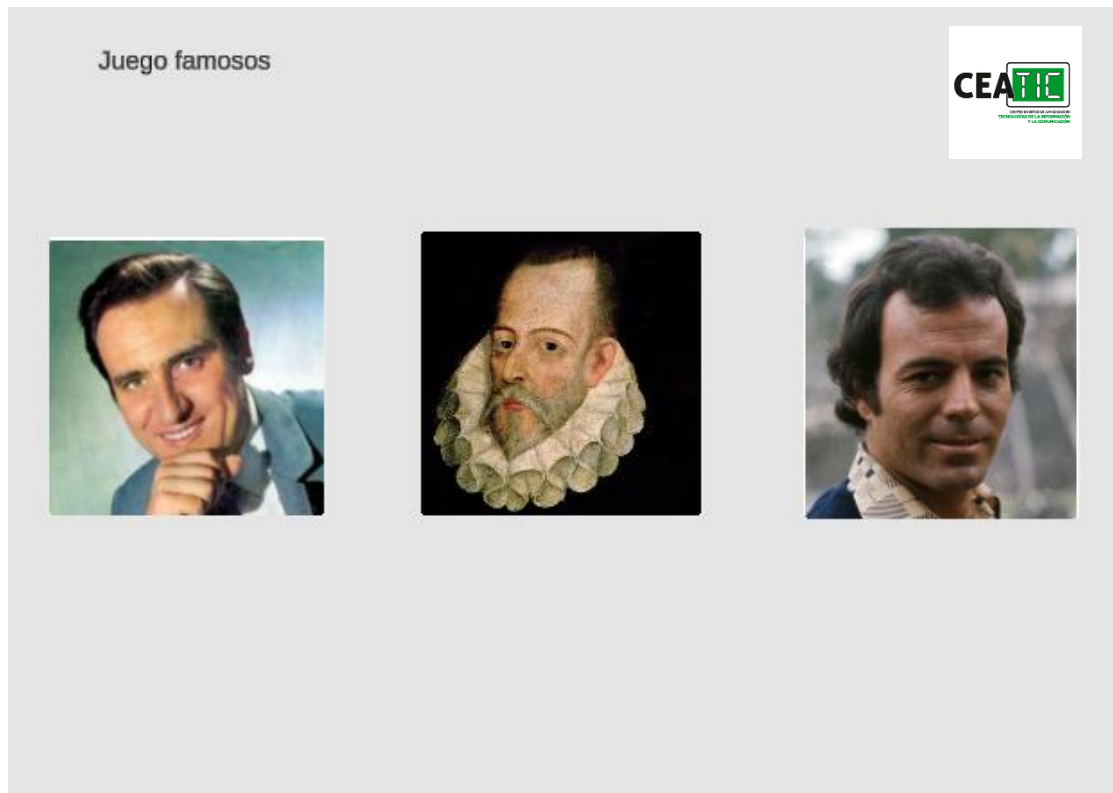


Ilustración 10 Storyboard pantalla famosos

El funcionamiento de los juegos diseñados es simple. Primero, empezaré explicando el juego de “Famosos”. Este juego de memoria consiste en reconocer a personajes famosos mediante citas o fragmentos de sus obras, como podrían ser una canción o un libro. Al inicio del juego, como en los demás, se dará una explicación de cómo jugar. En el primer turno, se pondrá el fragmento de la obra por el altavoz, y el mayor tendrá que reconocer el personaje al que pertenece. En el siguiente turno, se le preguntará por su profesión. Por último, en el tercer turno se enseñan a través de pantalla 3 imágenes de famosos, como se ve en la ilustración. El mayor deberá señalar cuál de los 3 es correcta. Una vez cumplida la secuencia de turnos, se repite 2 veces más con diferentes famosos.

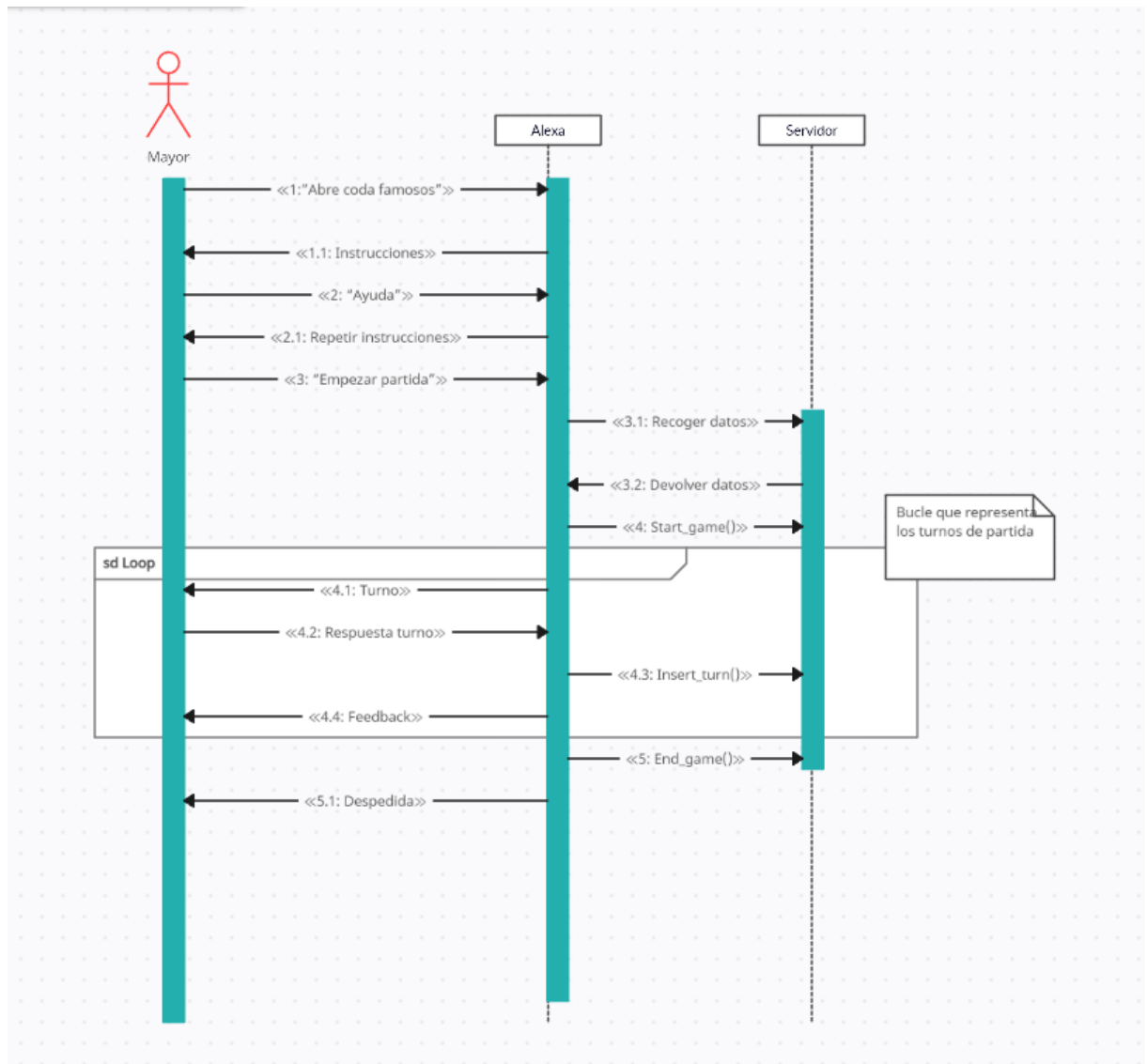


Ilustración 11 Diagrama de secuencia de Famosos

En cuanto al otro juego diseñado, “Secuencia numérica”, consiste en repetir la secuencia de números que el altavoz le presenta al mayor. Hay dos variantes, la directa y la indirecta, y ambas tienen su explicación dentro del juego. La directa simplemente consiste en repetir la secuencia en el mismo orden que el altavoz usó, mientras que en la secuencia indirecta debe ser el orden inverso. Las secuencias empiezan por 3 dígitos, aumentando 1 dígito cada dos secuencias. Si se fallan ambas secuencias con el mismo número de dígitos o se llega a la última secuencia de 9 dígitos, se termina la partida.

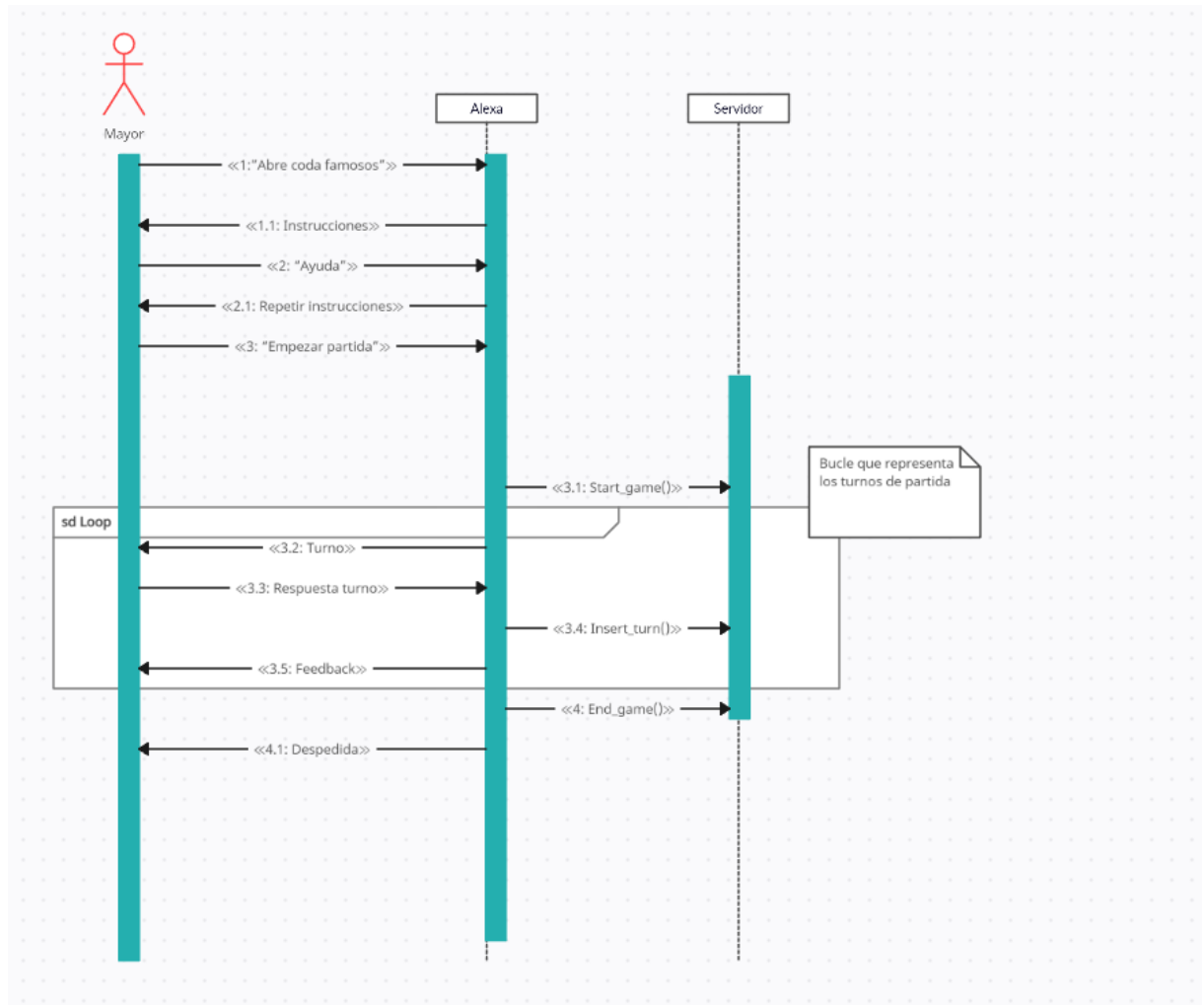


Ilustración 12 Diagrama de secuencia de Secuencia numérica

2.2.2 Análisis y diseño de la aplicación de Java

En este subapartado se expone el diseño de la interfaz del programa de Java y su funcionamiento básico. Como se puede apreciar, es una interfaz sencilla con 2 pestañas, que permiten filtrar los datos según el nombre del paciente o su correo asociado. En ambas hay un botón para hacer la búsqueda sin filtro. Su funcionamiento es simple: cuando se le da al botón de aceptar o sin filtro, genera un documento csv con los datos recogidos hasta la fecha de la base de datos. Si se hace con alguno de los filtros, será necesario rellenar los campos. Si no se rellenan, la aplicación dará una alerta avisando de este requerimiento.



CodaKoba

Nombre Correo

Nombre

Apellidos

Aceptar

Sin filtro

Detailed description: This storyboard shows a registration form. At the top left is the text 'CodaKoba'. Below it are two tabs: 'Nombre' (selected) and 'Correo'. Under the 'Nombre' tab, there are two text input fields: one labeled 'Nombre' and another labeled 'Apellidos'. At the bottom of the form are two buttons: 'Aceptar' and 'Sin filtro'.

Ilustración 13 Storyboard de aplicación java 1

CodaKoba

Nombre Correo

Correo

Aceptar

Sin filtro

Detailed description: This storyboard shows the same registration form as the previous one, but with the 'Correo' tab selected. The 'Nombre' tab is now inactive. The text input field labeled 'Correo' is now active, while the 'Nombre' and 'Apellidos' fields are hidden. The 'CodaKoba' text, 'Aceptar' button, and 'Sin filtro' button remain in the same positions.

Ilustración 14 Storyboard de aplicación java 2

2.2.3 Análisis y diseño de las bases de datos

En este subapartado, se expone el diseño de las bases de datos. En el proyecto se han usado dos bases de datos, una general llamada Codakoba, y una específica del juego de Famosos, llamada igual que el juego. Primero, nos centraremos en el diseño de la base de datos CodaKoba.

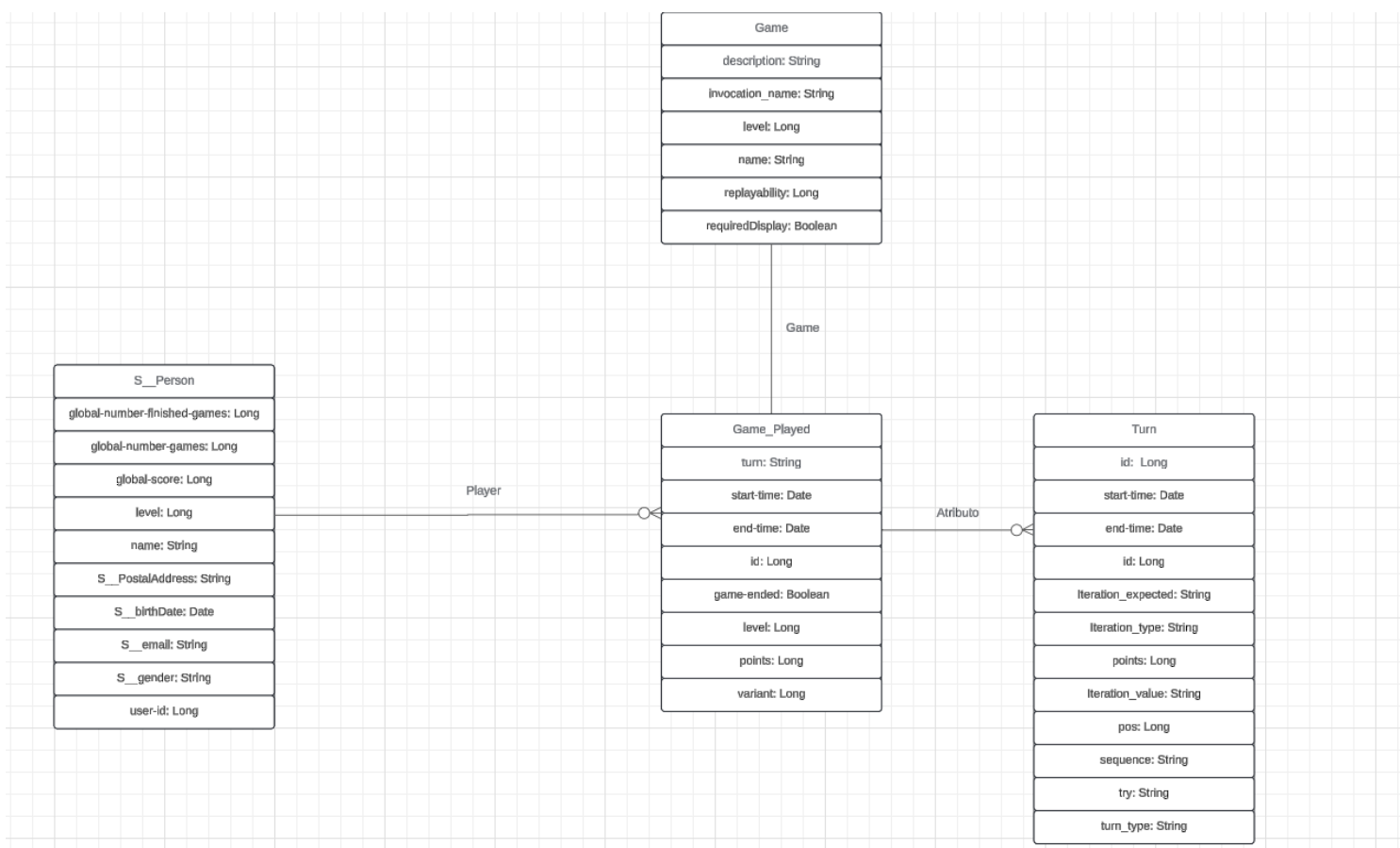


Ilustración 15 Esquema de la base de datos CodaKoba

- **S__Person**: Esta entidad es la encargada de guardar la información de los usuarios del sistema respectiva a su nombre y datos de contacto, nivel de dificultad máximo (para uso futuro) y estadísticas globales de juegos.
- **Game_Played**: Esta es una relación que enlaza a los usuarios (jugadores) con los juegos (games), dando lugar a partidas. Almacena los turnos jugados, la duración de la partida, el nivel de dificultad, la variante del juego y los puntos conseguidos.

- Turn: Es un atributo que almacena el tiempo de inicio y fin del turno jugado, la interacción del usuario, los puntos ganados ese turno, qué turno es dentro del juego, el tipo de interacción, el intento y la respuesta correcta.
- Game: Es una entidad que almacena la información de los juegos. Entre la información almacenada está su descripción, la frase de invocación, el nivel de dificultad, el nombre, su reusabilidad y si necesita o no pantalla táctil.

Un detalle importante de TypeDB que aún no se ha comentado, es que todas las entidades, atributos y relaciones, pueden contener a cualquiera de los otros tipos, ya que todas descienden de la clase Entity. Esto se hace aparente en el esquema de CodaKoba al ver que un atributo, turn, tiene a su vez varios atributos, o que la relación Game-played tiene a su vez atributos asignados. Ahora, se expondrá el esquema de Famosos.

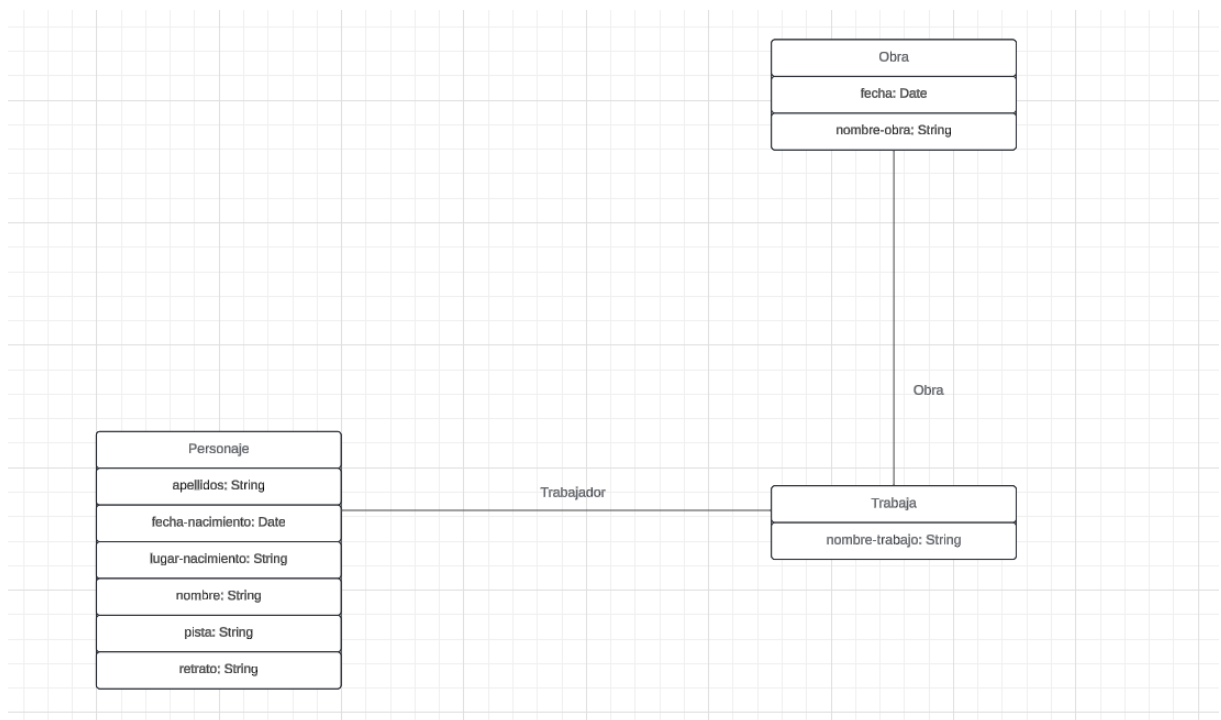


Ilustración 16 Esquema de la base de datos famosos

- Personaje: Esta entidad guarda información sobre los personajes famosos que aparecerán en el juego. Los atributos que contiene almacenan los apellidos, la fecha de nacimiento, el lugar de nacimiento, el nombre, la pista (url al audio de su obra) y su retrato (url a la imagen).

- Trabaja: Es una relación que enlaza al trabajador (personaje) con su obra más conocida (obra). Almacena un atributo con el nombre de la obra
- Obra: Es una entidad que almacena los datos de la obra del famoso. Sus atributos son la fecha de la obra y su nombre.

3 DESARROLLO

En este apartado, se especifica cada iteración o incremento del proyecto, que como se ha especificado anteriormente ha seguido en su desarrollo una metodología incremental. Los subapartados a continuación se corresponden con los incrementos especificados en la planificación temporal del proyecto.

3.1 Tarea 1: Aprendizaje de tecnologías

Durante la primera fase del proyecto, se dedicó el esfuerzo al aprendizaje de las tecnologías que serían usadas durante el mismo, detalladas en el apartado 1.11 del documento. Al ser un periodo de aprendizaje, no se desarrolló software, con lo cual no habrá subapartados de esta tarea.

3.2 Tarea 2: Diseño y desarrollo de las aplicaciones

Durante esta tarea, se diseñó y desarrolló los minijuegos de Famosos y Secuencia numérica, siguiendo las especificaciones que los psicólogos proporcionaron. Al no tener aún base de datos, solo se desarrolló en esta etapa el código del funcionamiento del juego como tal.

El funcionamiento de los juegos consistía en reconocer la frase de invocación de la skill, de lo que se encargaba la propia Alexa, cargar los datos de prueba (urls a imágenes y audios predefinidos en el caso de famosos) y comenzar el bucle de juego. Para crear este bucle, se sobrescribió la lógica que maneja Alexa para seleccionar Intents, redirigiendo directamente a los Intents que necesitábamos. Esto se hizo mediante el uso de directivas, concretamente la ElicitSlot directive [25], que permite asignar el Slot por el que queremos que Alexa pregunte, y a qué Intent debe dirigirse

el código. Una vez se terminaba el juego (terminando los turnos en Famosos y Secuencia numérica o acumulando 3 fallos en este último), se finalizaba la partida.

3.2.1 Pruebas de verificación del sistema

Para verificar el funcionamiento del sistema, se hicieron varias partidas en el dispositivo, que permitieron ir depurando el código una vez implementado. Gracias al servicio de logs de Amazon se pudo depurar el código correctamente, eliminando cualquier error que pudiera surgir durante la partida.

3.2.2 Pruebas de validación del sistema

Para validar los juegos desarrollados, se le dio acceso al equipo de psicólogos a los minijuegos mediante un beta test, asignándoles como testers. Una vez se comprobó la adecuación de los juegos, se pasó al diseño de la base de datos.

3.3 Tarea 3: Diseño de la base de datos

Una vez finalizado el prototipado de las skills, se diseñaron e implementaron las bases de datos necesarias para su funcionamiento. En un inicio, se usó un servidor en AWS (Amazon Web Services) para hospedar la base de datos y los recursos multimedia, pero finalmente se cambió a un servidor del CEATIC, usando un servidor web para el contenido multimedia. El esquema completo de las bases de datos ha sido añadido al final del documento en los apéndices.

También hay que comentar que, pese a que en el diagrama de Gantt las tareas 2 y 3 aparecen secuencialmente, una gran parte del desarrollo se hizo de manera concurrente, ya que no interferían entre si.

3.3.1 Pruebas de verificación del sistema

En esta fase no hubo ninguna prueba de verificación, ya que, si hay algún error a la hora de diseñar el esquema o poblar la base de datos con los datos iniciales, TypeDB da un error y cierra la transacción.

3.3.2 Pruebas de validación del sistema

Durante esta fase se trabajó con los psicólogos para reunir las especificaciones de los datos que sería necesario almacenar en la base de datos.

3.4 Tarea 4: Conexión de las aplicaciones con Backend

En esta fase, se completó el funcionamiento inicial de las aplicaciones conectándolas a la base de datos y el servidor web del CEATIC. Para ello, se usó la biblioteca de TypeDB para Python y se diseñó un documento de formato .py de uso general para las aplicaciones, reuniendo las consultas comunes como la transacción para iniciar una partida, finalizarla o guardar los turnos. Las consultas específicas, como las del juego Famosos, que carga datos específicos de su funcionamiento, se añadieron al final de dicho documento dentro del juego. En la siguiente ilustración se muestra un ejemplo de las transacciones diseñadas.

```
@connect_to_db
def read_transaction(client, db: str, query: str, values: List):
    """Makes a read query to the specified TypeDB session.

    Args:
        client:
            The database client as provided by the decorator.
        db:
            The name of the TypeDB database to query.
        query:
            The actual query.
        values:
            A list of concepts (str) that will be extracted from the query.

    Returns:
        A list with every concept matching the query.
    """
    logger.debug("Leyendo de la base de datos")
    with client.session(db, SessionType.DATA) as session:
        with session.transaction(TransactionType.READ) as transaction:
            it = transaction.query().match(query)
            return [ans.get(value).get_value() for ans in it for value in values]

def get_user_name(user_email: str) -> str:
    """Looks for the username associated with a given user id.

    Args:
        user_email:
            An email as extracted from the Alexa's session.

    Returns:
        A string containing the user name in natural language.
    """
    logger.debug("Recogiendo nombre de usuario")
    query = ""
    query = query.join([
        'match',
        ' $person isa s__Person, has s__email "{}";'.format(user_email),
        ' $name isa s__additionalName;',
        ' $person has $name;',
        'get $name;'
    ])

    #try:
    respuesta = read_transaction(db='codakoba', query=query, values=['name'])[0]
    #except:
    #raise Exception("No existe el email en la base de datos. Por favor, póngase en contacto con el administrador")

    #respuesta = read_transaction(db='codakoba', query=query, values=['name'])[0]
    return respuesta
```

Ilustración 17 Transacción get_user_name

3.4.1 Pruebas de verificación del sistema

Para verificar el funcionamiento de las consultas, se jugaron varias partidas, asegurándose tras cada partida que las partidas se guardaban correctamente. Para revisar que no hubiera problemas, se recorrieron las diferentes casuísticas que se pueden dar en una partida, tales como finalizarla abruptamente, ganar o perder el juego o iniciar la skill sin iniciar partida.

3.4.2 Pruebas de validación del sistema

Para validar el sistema, se revisó la base de datos para comprobar la existencia de los datos que debían insertarse durante las pruebas de verificación. Una vez comprobado que todos los datos necesarios eran guardados correctamente, se pasó a la siguiente tarea.

3.5 Tarea 5: Inclusión de APL

Para mejorar la experiencia de usuario, se decidió hacer uso de APL para estimular al mayor con imágenes. Durante esta tarea, se amplió la funcionalidad de los juegos para mostrar imágenes por la pantalla del dispositivo, añadiendo una pantalla por defecto y, en el caso del juego de famosos, un turno entero que consiste en discernir la imagen del famoso con el que se está jugando. Cada pantalla consiste en un documento APL. Un ejemplo de documento APL se muestra en la

siguiente ilustración.

```
1 {  
2   "type": "APL",  
3   "version": "1.6",  
4   "theme": "dark",  
5   "import": [  
6     {  
7       "name": "alexa-layouts",  
8       "version": "1.3.0"  
9     }  
10  ],  
11  "mainTemplate": {  
12    "parameters": [  
13      "imageListData"  
14    ],  
15    "items": [  
16      {  
17        "type": "AlexaImageList",  
18        "id": "famousList",  
19        "backgroundColorOverlay": false,  
20        "imageAspectRatio": "square",  
21        "imageMetadataPrimacy": true,  
22        "imageScale": "best-fill",  
23        "listItems": "${imageListData.listItems}",  
24        "headerAttributionImage": "https://pbs.twimg.com/profile_images/1440679180362928133/nUCMyuCm_400x400.jpg",  
25        "headerAttributionText": "Sistemas inteligentes de acceso a la información",  
26        "headerDivider": true,  
27        "primaryAction": [  
28          {  
29            "type": "SendEvent",  
30            "arguments": ["PictureSelected",  
31                          "${ordinal}"]  
32          }  
33        ]  
34      }  
35    ]  
36  }  
37 }  
38 }
```

Ilustración 18 APL táctil

Como se puede apreciar, es un documento con estructura Json, que coge una lista de imágenes y las dispone en una fila. También se usan atributos de header para añadir el icono del proyecto y un título. Por último, al final del documento se puede ver una acción que envía un evento al event listener de Alexa cuando se toca en una de estas imágenes.

3.5.1 Pruebas de verificación del sistema

Para comprobar el funcionamiento correcto de los documentos APL, bastó con jugar una partida a cada juego para comprobar que se mostraran las imágenes correctamente y en posiciones adecuadas, ya que se cargan mediante urls.

3.5.2 Pruebas de validación del sistema

En esta tarea, las pruebas de validación consistieron en mostrar los resultados a los psicólogos, que dieron feedback sobre pequeños cambios en la disposición de imágenes y otros aspectos menores. Una vez solucionados, se empezaron a pasar pruebas reales con mayores.

3.6 Tarea 6: Aplicación de Java

Para permitir a los psicólogos acceder a los datos en cualquier momento sin tener que acceder a la base de datos manualmente, se diseñó un pequeño programa en Java, usando la librería de TypeDB para este lenguaje, que permitía hacer consultas a la base de datos que sacaran los datos de los Game_played, teniendo como filtro el nombre del usuario o su correo, pudiéndose hacer también sin filtro. En la siguiente ilustración se muestra un ejemplo de salida de la aplicación.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	DIA	HORA	Secuencia	Nº de ensayo R dada	Puntos	R correcta	Tiempo de reacción en msag (tiempo que va desde que Alexa finaliza la secuencia hasta que el sujeto comienza a responder)															
2	20/07/2022	8:07:00	3	2	582	0 [5, 8, 2]	8															
3	01/09/2022	8:09:00	3	1	582	1 [5, 8, 2]	13															
4	01/09/2022	8:09:00	3	1	694	1 [6, 9, 4]	8															
5	01/09/2022	9:09:00	3	2	4596	0 [5, 8, 2]	3															
6	01/09/2022	9:09:00	3	2	3	0 [6, 9, 4]	5															
7	01/09/2022	9:09:00	3	2	3	0 [6, 9, 4]	5															
8	01/09/2022	9:09:00	3	2	258	0 [5, 8, 2]	6															
9	01/09/2022	9:09:00	3	1	496	1 [6, 9, 4]	7															
10	01/09/2022	9:09:00	4	2	6439	0 [6, 4, 3, 9]	6															
11	25/10/2022	8:10:00	3	2	582	0 [5, 8, 2]	5															
12	25/10/2022	8:10:00	3	2	594	0 [6, 9, 4]	7															
13	25/10/2022	8:10:00	3	2	594	0 [6, 9, 4]	7															
14	25/10/2022	8:10:00	3	2	582	0 [5, 8, 2]	5															
15	25/10/2022	8:10:00	3	2	594	0 [6, 9, 4]	48															
16	25/10/2022	8:10:00	3	2	594	0 [6, 9, 4]	48															
17	25/10/2022	8:10:00	3	1	582	1 [5, 8, 2]	7															
18	25/10/2022	8:10:00	3	1	694	1 [6, 9, 4]	9															
19	08/11/2022	18:11:00	0	1	calle maestr	33 Calle Maestr	57															
20	08/11/2022	18:11:00	1	1	calle virgilio	33 Calle Virgilio	18															
21	08/11/2022	18:11:00	2	1	plaza de san	33 Plaza de San	12															
22	08/11/2022	19:11:00	0	1	calle de los c	33 Calle Los Cai	17															
23	08/11/2022	19:11:00	1	1	calle los cañ	33 Plaza de los	11															
24	08/11/2022	19:11:00	0	1	calle de ubei	33 Calle de Ube	12															
25	08/11/2022	19:11:00	1	1	calle virgen c	33 Calle Virgen	14															
26	08/11/2022	19:11:00	0	1	plaza de las	33 Plaza de las	11															
27	08/11/2022	19:11:00	1	1	virgen de la c	33 Calle Virgen	12															
28	08/11/2022	19:11:00	2	1	explanada d	33 Explanada d	17															
29	09/11/2022	8:11:00	2	2	amarillosos	1 amarillosos	7															
30	09/11/2022	8:11:00	3	1	rosaverde	1 rosaverde	6															
31	03/01/2023	17:01:00	2	2		0 busbici	29															
32	03/01/2023	17:01:00	2	2		0 busbici	31															
33	03/01/2023	17:01:00	2	2		0 avióncoche	21															
34	03/01/2023	17:01:00	2	2		0 trenbici	22															
35	03/01/2023	17:01:00	2	2	aviónavión	0 trenbici	22															
36	03/01/2023	17:01:00	2	2		0 avióncoche	22															

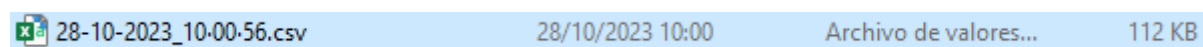


Ilustración 19 Formato del fichero CSV

3.6.1 Pruebas de verificación del sistema

Para esta tarea, la verificación consistió en la generación de diferentes documentos, usando los filtros especificados, antes y después de generar nuevos datos, asegurándose que estos se reflejaban correctamente.

3.6.2 Pruebas de validación del sistema

Al ser el equipo de psicólogos el que nos especificó el formato de csv necesario, y siendo una aplicación simple, no fue necesario hacer una validación de esta parte del sistema.

3.7 Pruebas finales

Este apartado corresponde a la tarea 7, donde se realizaron cambios menores a la aplicación para facilitar el uso. Se realizaron durante las pruebas con personas mayores, atendiendo al feedback de los mismos. A continuación, se expone una lista de estos cambios y el resultado final, mediante una serie de imágenes de los juegos funcionando.

- Normalizar el volumen entre audios
- Añadir numeración a las imágenes
- Cambiar secuencia numérica de números aleatorios a secuencia específica
- Cambiar condiciones de derrota en el juego de secuencia numérica
- formato específico de CSV para fichero de salida
- Añadir alerta de inicio de turno
- Añadir una explicación al inicio del juego

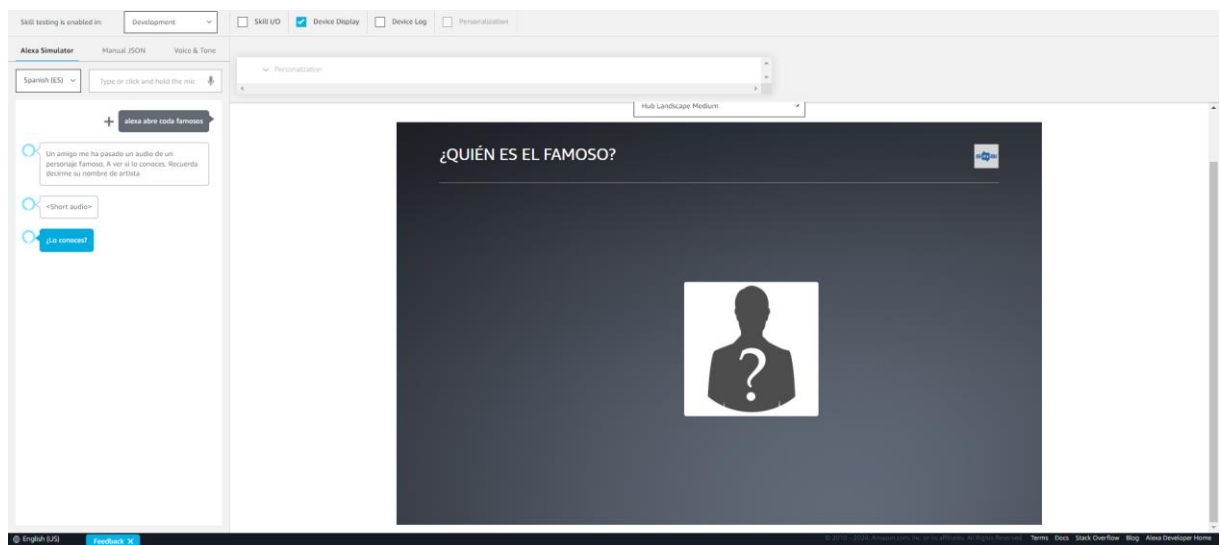


Ilustración 20 Juego Famosos 1

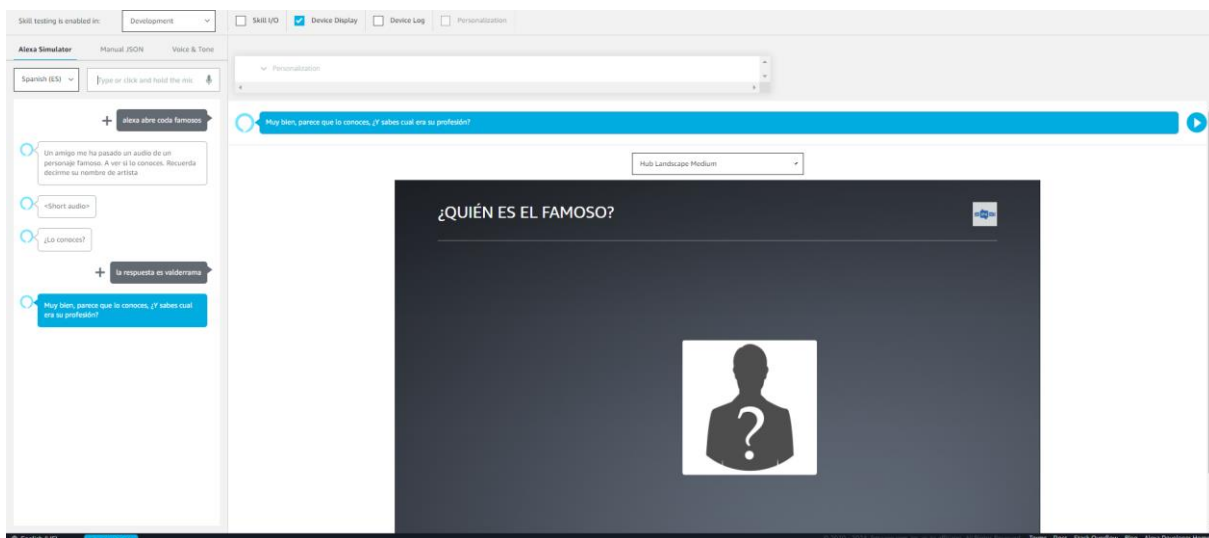


Ilustración 21 Juego Famosos 2

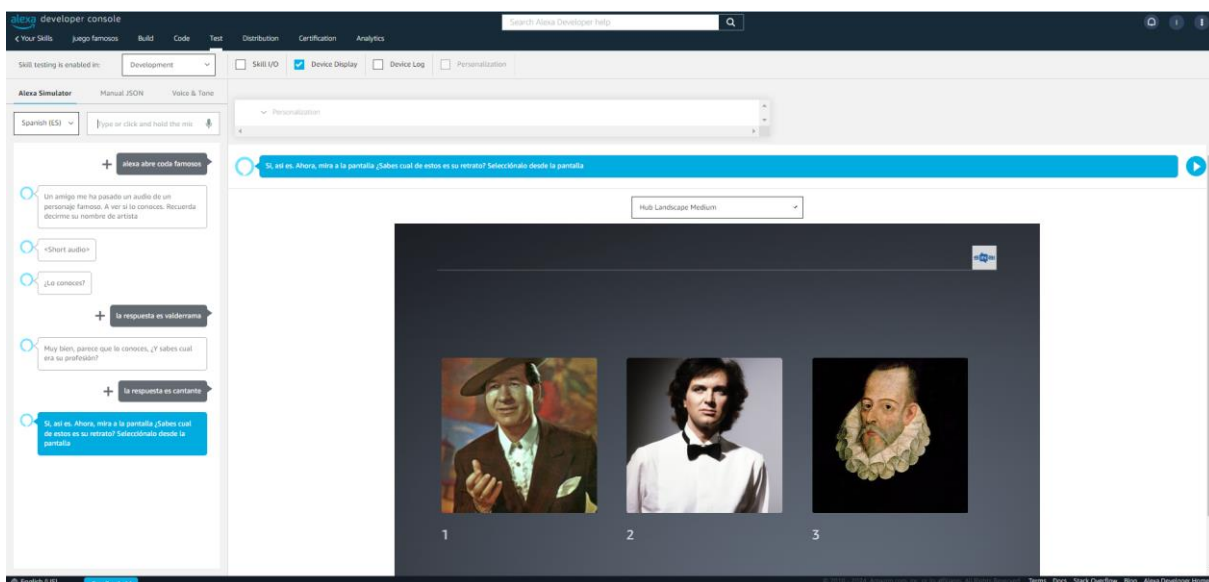
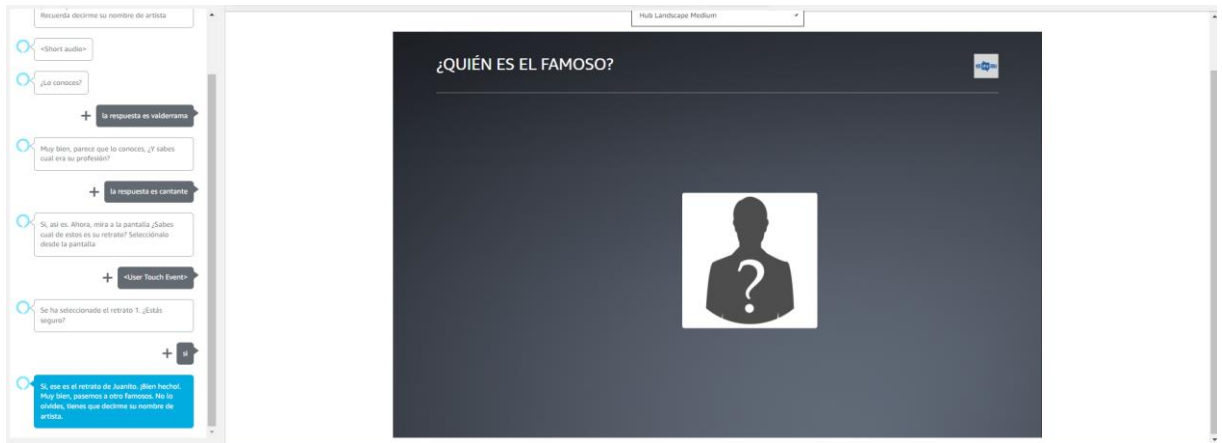
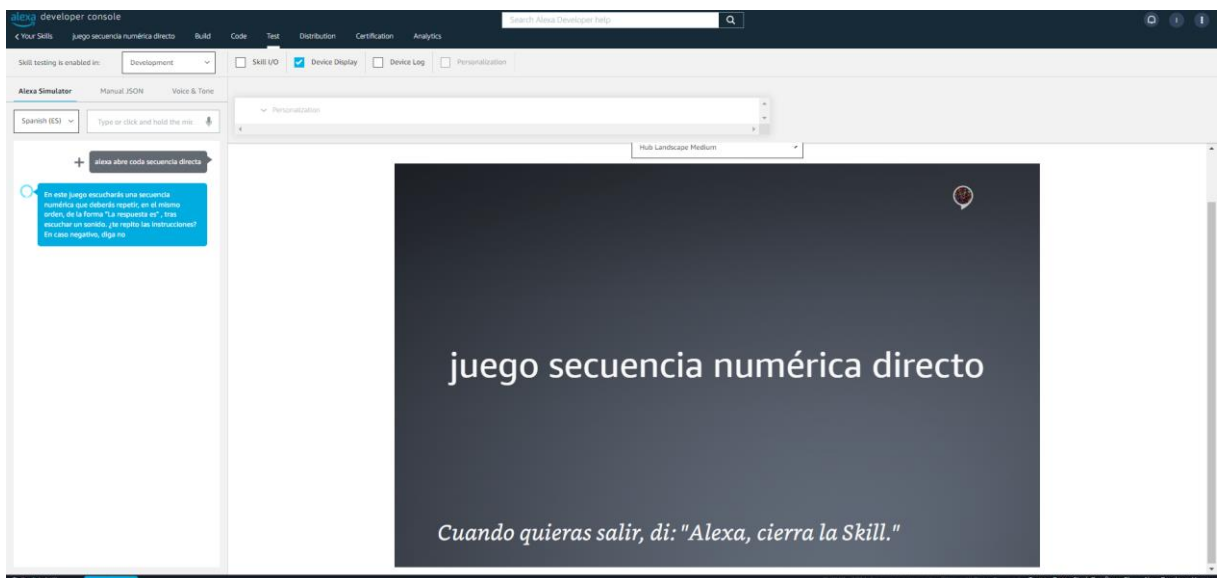


Ilustración 22 Juego Famosos 3

*Ilustración 23 Juego Famosos 4**Ilustración 24 Juego Secuencia numérica 1*

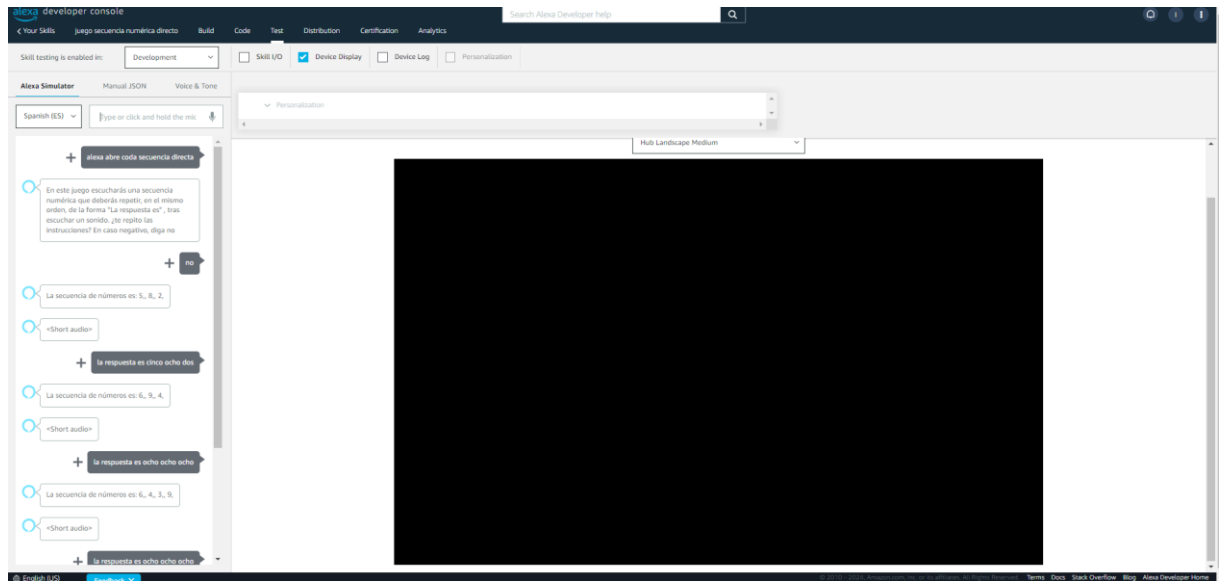


Ilustración 25 Secuencia numérica 2

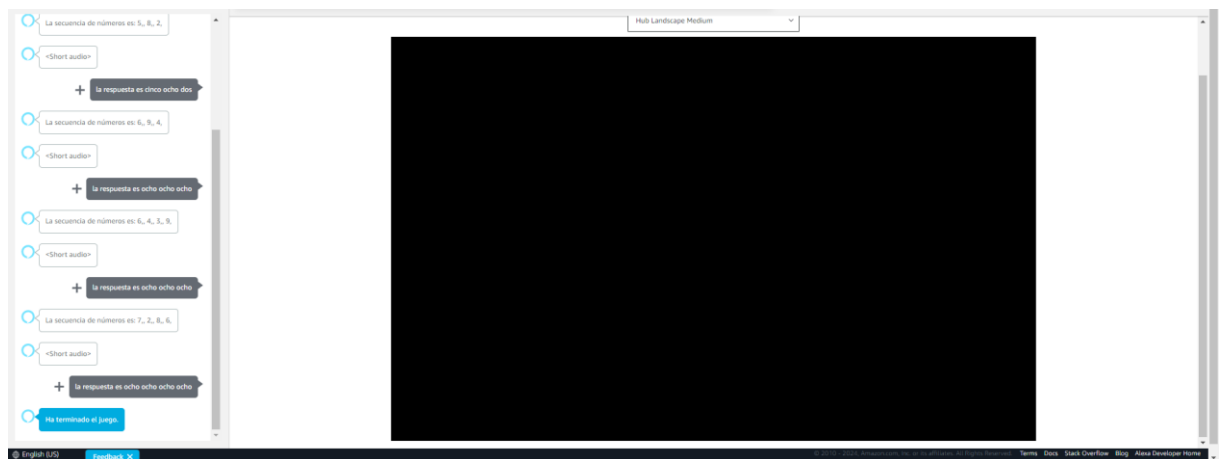


Ilustración 26 Secuencia numérica 3

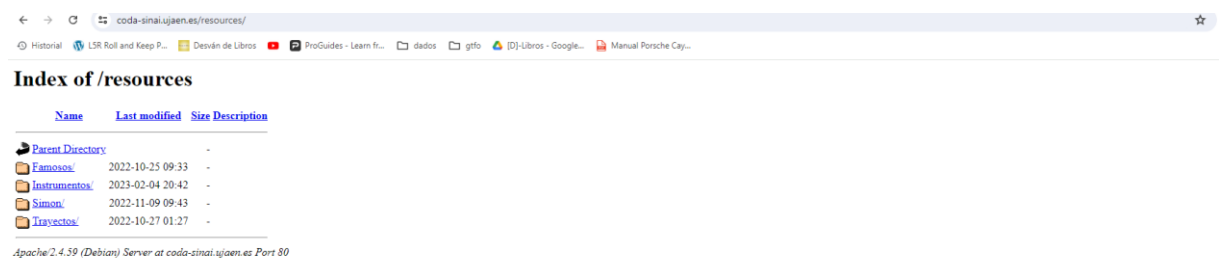
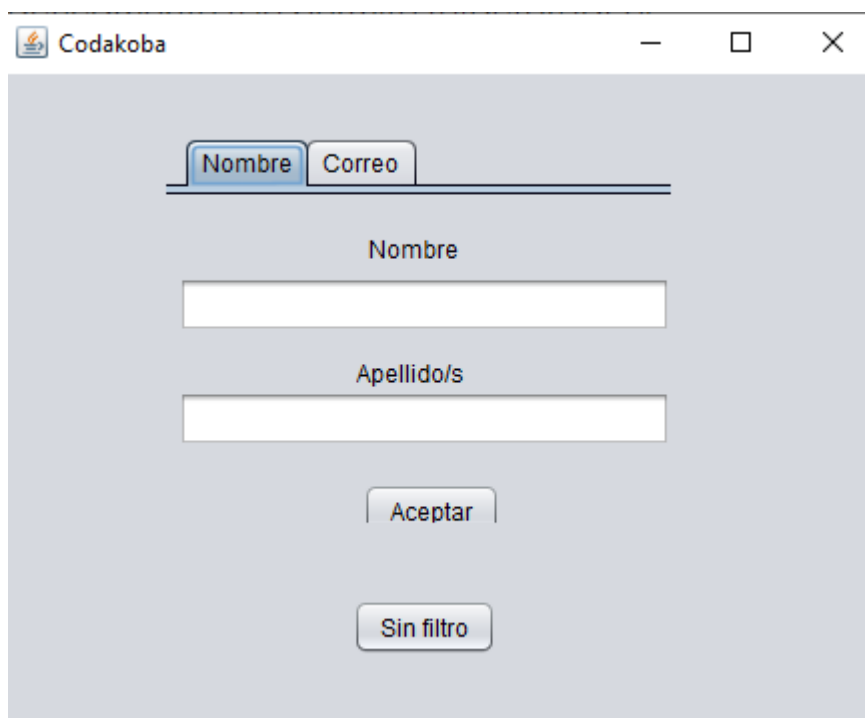
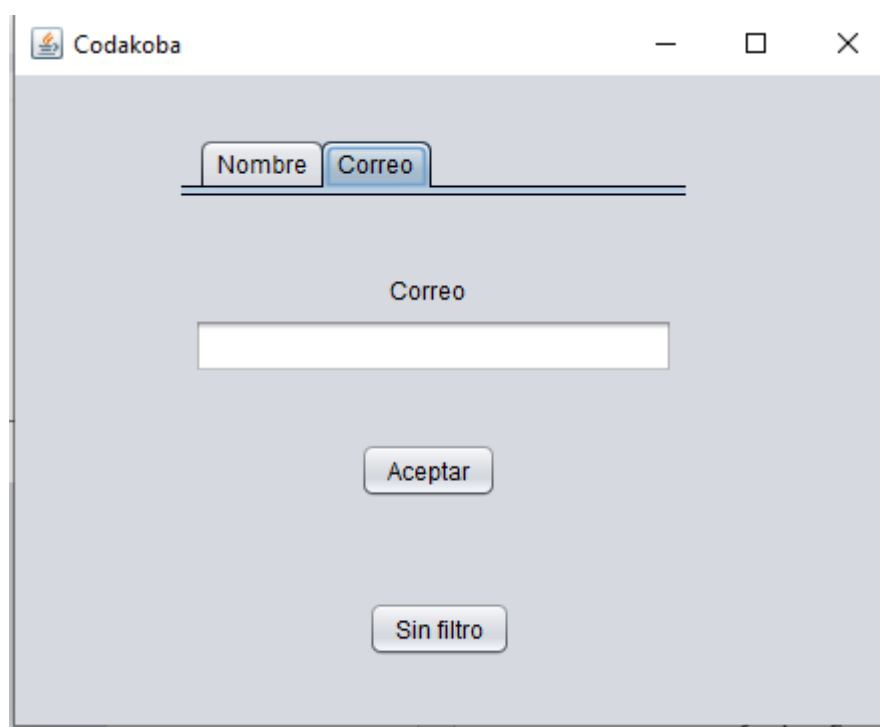


Ilustración 27 Servidor Web



The screenshot shows a Java application window titled 'Codakoba'. At the top, there are two tabs: 'Nombre' (selected) and 'Correo'. Below the tabs, the label 'Nombre' is centered above a text input field. Below this, the label 'Apellido/s' is centered above another text input field. At the bottom, there are two buttons: 'Aceptar' and 'Sin filtro'.

Ilustración 28 Pantalla de nombre - Aplicación Java



The screenshot shows the same Java application window 'Codakoba', but the 'Correo' tab is now selected. The 'Nombre' tab is still visible. The label 'Correo' is centered above a single text input field. The 'Aceptar' and 'Sin filtro' buttons remain at the bottom.

Ilustración 29 Pantalla de correo – Aplicación Java

4 CONCLUSIONES Y TRABAJOS FUTUROS

Para finalizar con este trabajo, hablaré de las conclusiones sacadas del proyecto y los planes a futuro que hay para continuar con este. Como se puede comprobar, hemos conseguido una aplicación completa que permite a los psicólogos sacar información sobre la memoria del paciente sin necesidad de interacción humana, con el añadido de ser completamente transparente y, al tratarse de minijuegos, un entretenimiento para el mismo. Con las diferentes interacciones que se ha tenido con el equipo de psicólogos, hemos mejorado los juegos para que sean claros en su explicación y fáciles de manejar para las personas mayores. Por desgracia, las limitaciones impuestas por el dispositivo (Alexa), como la restricción de 8 segundos para dar una respuesta, han limitado la funcionalidad de estos. Gracias a las bases de datos desarrolladas en el proyecto, podemos guardar un histórico de las interacciones del usuario con los minijuegos. Por último, se ha desarrollado un programa fácil de utilizar que permitirá a los psicólogos consultar los datos obtenidos en cualquier momento.

En cuanto a los planes a futuro, se planea continuar al proyecto, añadiendo minijuegos y ampliando las funcionalidades de estos. La base de datos se diseñó inicialmente con objetivo de hospedar información para realizar también pequeñas conversaciones con el paciente personalizables a través de los datos, como familiares o gustos. También se planea hacer un sistema de alerta que avise al mayor para que juegue al juego si no ha jugado durante un periodo de tiempo, así como un sistema de retos que fomente el uso de las aplicaciones.

5 DEFINICIONES Y ABREVIATURAS

- Memoria de trabajo: “Un sistema cerebral que proporciona almacenamiento temporal y manipulación de la información necesaria para tareas cognitivas complejas, como la comprensión del lenguaje, el aprendizaje y el razonamiento” [26]
- TIC: Abreviatura de Tecnologías de la Información y la Comunicación. Conjunto de recursos

- Serious games: Juegos diseñados con un propósito distinto a la diversión, normalmente orientados al aprendizaje
- Grafos de propiedades: “Un grafo de propiedades tiene vértices que pueden contener información detallada sobre un tema, así como bordes para resaltar la relación entre esos vértices. Los vértices y los bordes pueden tener atributos (llamados propiedades) a los que están asociados” [27]
- CEATIC: Centro de Estudios Avanzados en TIC
- NoSQL: Not only SQL. “Enfoque utilizado en el diseño de bases de datos que permite el almacenamiento y consulta de datos fuera de las estructuras tradicionales que se encuentran en las bases de datos relacionales” [14]
- SQL: Structured Query Language. “Lenguaje de programación para almacenar y procesar información en una base de datos relacional” [15]
- Base de datos relacional: “Base de datos que almacena información en forma de tabla, con filas y columnas que representan diferentes atributos de datos y las diversas relaciones entre los valores de datos” [15]
- SSL: Secure Socket Layer. “Sistema que cifra automáticamente la información antes de enviarla a través de Internet y la descifra en el otro extremo antes de utilizarla. Con estas operaciones se protege la información confidencial como, por ejemplo, los dígitos de las tarjetas de crédito, mientras se transmite a través de Internet” [16]
- MP3: MPEG-2 Audio Layer III
- MPEG: Moving Picture Experts Group
- Skill de Alexa: Las skills de Alexa son aplicaciones activadas por voz que añaden capacidades al dispositivo.
- ASK: Alexa Skill Kit. “Conjunto de herramientas, documentación, muestras de código y API en self-service con el que puedes añadir Skills a Alexa de forma rápida y sencilla. El ASK permite a diseñadores, desarrolladores y marcas crear Skills atractivas y llegar a los consumidores” [17]
- IDE: Entorno de desarrollo integrado

- APL: Alexa Presentation Language. Objeto JSON que define una plantilla que mostrar por la pantalla de un dispositivo Alexa [18]
- JSON: JavaScript Object Notation. Formato ligero de intercambio de datos [19]
- BOE: Boletín Oficial del Estado
- Codakoba: Nombre del proyecto
- Slot: Variable fuertemente tipada que puede ser usada a lo largo de toda la aplicación
- Intent de Alexa: Representa una acción que cumple con la orden de voz
- AWS: Amazon Web Services

6 BIBLIOGRAFÍA

1. Diagnóstico Eje Envejecimiento, Gobierno de España. Consultado el 24/05/2023: https://www.miteco.gob.es/content/dam/mitesco/es/retodemografico/temas/analisis-cartografia/diagnostico_eje_envejecimineto_tcm30-517770.pdf
2. Asistente virtual de Google. Consultado el 28/05/2023: https://assistant.google.com/intl/es_es/
3. Asistente virtual de Amazon: Alexa. Consultado el 28/05/2023: https://developer.amazon.com/en-US/alexa?ref=MARS_NAV_desktop_help_rscs_create_tog_Alx_for_Dev
4. Asistente virtual de Microsoft: Cortana. Consultado el 28/05/2023: <https://support.microsoft.com/es-es/topic/-qu%C3%A9-es-cortana-953e648d-5668-e017-1341-7f26f7d0f825>
5. Asistente virtual de Samsung: Bixby. Consultado el 28/05/2023: https://www.samsung.com/es/apps//bixby/#bixby_for_everyone
6. Estudio sobre biomarcadores en la enfermedad del Alzheimer. Consultado el 29/05/2023: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10197768/>

7. LETHE (λήθη) – A personalized prediction and intervention model for early detection and reduction of risk factors causing dementia, based on AI and distributed Machine Learning. Consultado el 29/05/2023: <https://cordis.europa.eu/project/id/101017405>
8. Multicentric Language Markers of NeuroDegeneration. Consultado el 29/05/2023: <https://cordis.europa.eu/project/id/101025814>
9. Intelligent digital tools for screening of brain connectivity and dementia risk estimation in people affected by mild cognitive impairment. Consultado el 29/05/2023: <https://cordis.europa.eu/project/id/964220>
10. MRI analysis tool for early detection of dementia. Consultado el 29/05/2023: <https://cordis.europa.eu/project/id/101113566>
11. Base de datos MongoDB. Consultado el 02/06/2023: <https://www.mongodb.com/>
12. Base de datos TypeDB. Consultado el 02/06/2023: <https://typedb.com/>
13. Requisitos de seguridad para contenido multimedia en Skills de Alexa. Consultado el 04/05/2023: <https://developer.amazon.com/en-US/docs/alexa/alexa-presentation-language/apl-image.html>
14. ¿Qué es NoSQL? Consultado el 04/06/2023: <https://www.ibm.com/es-es/topics/nosql-databases>
15. ¿Qué es SQL? Consultado el 04/06/2023: <https://aws.amazon.com/es/what-is/sql/>
16. ¿Qué es SSL? Consultado el 04/06/2023: https://www.ibm.com/docs/es/was-nd/8.5.5?topic=SSAW57_8.5.5/com.ibm.websphere.edge.doc/cp/admingd36.htm
17. ¿Qué es ASK? Consultado el 05/06/2023: <https://developer.amazon.com/es-ES/alexa/alexa-skills-kit>
18. ¿Qué es APL? Consultado el 06/06/2023: <https://developer.amazon.com/en-US/docs/alexa/alexa-presentation-language/apl-document.html>

19. ¿Qué es JSON? Consultado el 06/06/2023: <https://www.json.org/json-es.html>
20. Paquetes Office 365. Consultado el 12/06/2023: <https://www.microsoft.com/en-us/microsoft-365/buy/compare-all-microsoft-365-products?=&activetab=tab:primaryr1>
21. Calculadora de servicios de Amazon. Consultado el 12/06/2023: <https://calculator.aws/#/estimate>
22. Precio del dispositivo Alexa Echo Show 10. Consultado el 12/06/2023: <https://www.amazon.es/nuevo-echo-show-10-pantalla-inteligente-hd-con-movimiento-y-alexa-antracita/dp/B084P3KP2S>
23. BOE núm. 131 de 02/06/2023. Consultado el 12/06/2023: https://www.boe.es/diario_boe/txt.php?id=BOE-A-2023-13182
24. BOE núm. 40 de 16/02/2023. Consultado el 12/06/2023: https://www.boe.es/diario_boe/txt.php?id=BOE-A-2023-4219
25. Interfaz de diálogo de Alexa. Consultado el 18/06/2023: <https://developer.amazon.com/en-US/docs/alexa/custom-skills/dialog-interface-reference.html>
26. Working memory and learning: Contributions of Neuropsychology, Magdalena López, Doctora en psicología de la Universidad Católica Argentina. Consultado el 02/07/2024: <https://dialnet.unirioja.es/descarga/articulo/4853443.pdf>
27. Definición de grafos de propiedades. Consultado el 02/07/2024: <https://www.oracle.com/es/autonomous-database/what-is-graph-database/#:~:text=Un%20grafo%20de%20propiedades%20tiene,a%20los%20que%20est%C3%A1n%20asociados.>
28. Tabla de amortización lineal. Consultado el 02/07/2024: <https://sede.agenciatributaria.gob.es/Sede/ayuda/manuales-videos-folletos/manuales-practicos/irpf-2020/capitulo-7-rendimientos-actividades-economicas-directa/fase-1-determinacion-rendimiento-neto/amortizaciones-dotaciones-ejercicio-fiscalmente-deducibles/requisitos-generales/coeficientes-amortizacion-lineal.html>