

PROYECTO FIN DE CICLO

QuizCard

TUTOR INDIVIDUAL : José Luis Arias Cobreros

TUTORA COLECTIVA : M^a Alejandra Tomás Fernández

AUTORA : Arancha Fernández Argüelles

Curso: 2020 – 2021

Capítulo 1 . MEMORIA DEL PROYECTO

1.1. RESUMEN DE LA MOTIVACIÓN

La idea originaria de la aplicación QuizCard surge del **interés personal** por practicar la lengua inglesa de una manera sencilla, rápida y amena. Por ello se elige un formato de juego con tarjetas con diferentes modalidades (elección entre respuestas múltiples, writing, listening, o mero repaso).

Aunque existen en el mercado numerosas aplicaciones para la práctica del inglés y de idiomas en general, siendo en general mucho más elaboradas que esta, resultan en ocasiones demasiado amplias en su funcionamiento resultando menos ágiles en su uso.

El motivo último de la aplicación es ayudar a que cualquier usuario con interés por practicar uno o más idiomas extranjeros pueda disponer de una **herramienta lo más simple posible** en que apoyarse para conseguirlo.

1.2. OBJETIVOS Y ALCANCE DEL PROYECTO

El **objetivo** es crear una **herramienta sencilla** para el estudio y repaso de vocabulario y expresiones en idiomas extranjeros, siguiendo una filosofía de ‘abrir y jugar’ sin más pasos intermedios que distraigan al usuario del objetivo de revisar sus conocimientos.

Para hacer el uso de la aplicación lo más accesible posible **independientemente de la ocasión** se elige el formato web app, lo que permite dedicarle unos minutos al estudio en cualquier momento y en cualquier lugar (durante una espera, un viaje de trayecto corto, un descanso, ...).

Con objeto de hacer el proceso de aprendizaje más **ameno**, y de hacer del estudio de un idioma un **reto de equipo**, se elige un formato juego en grupo. Así se consigue que los avances ajenos ayuden al usuario a mantener la motivación por continuar.

Con este proyecto se pretende **alcanzar** el desarrollo total de una aplicación de juego 100% funcional con 4 modalidades de juego:

- Pregunta (escrita) con respuesta múltiple.
- Pregunta (escrita) a responder de forma escrita.
- Pregunta audible a responder de forma escrita
- Repaso con pregunta (escrita) a responder de forma escrita. La diferencia con la segunda modalidad consiste en que en modo repaso las respuestas, tanto aciertos como fallos, no se toman en cuenta en las estadísticas de los resultados del usuario.

No entrará dentro del alcance de este proyecto el registro y autenticación de usuarios. La gestión de usuarios podría implementarse haciendo uso de cualquier tecnología o plataforma, puesto que el desarrollo front-end de la web app permite la integración con cualquier modalidad de back-end elegida para ese fin.

A **nivel técnico** se busca ampliar conocimientos en dos aspectos en tendencia en el mercado laboral actual: el **desarrollo web** y las **bases de datos no relacionales**.

Se profundizará en el dominio de **TypeScript** y **Angular** como tecnologías para el desarrollo web front-end.

Se empleará la BD NoSQL **Firestore** de Google Cloud. Las BBDD documentales no fueron abordadas en la asignatura de *Bases de Datos* del primer curso, que se centraba exclusivamente en aquellas de tipo relacional, pero sí fueron introducidas en el segundo curso del ciclo.

Capítulo 2 . INTRODUCCIÓN

2.1. JUSTIFICACIÓN DEL PROYECTO

Este proyecto pretende profundizar en los contenidos de mayor tendencia en el panorama tecnológico actual tratados en las asignaturas de [Acceso a Datos](#) y [Programación Multiplataforma y Dispositivos Móviles](#): las bases de datos documentales y las aplicaciones web basadas SPA (Single Page Application).

Se hace la elección de una [base de datos documental](#) (o no relacional) en base a los aspectos:

- La aplicación está diseñada para llegar a manejar grandes volúmenes de registros (colecciones de cartas en forma de mazos y estadísticas) y consultar su recuento en tiempo real: en el número de cartas de un mazo, en el número de respuestas realizadas, en el número de aciertos/errores, en el número de usuarios que comparten un mazo o carta...
- El modelado de las estructuras de datos es de una gran sencillez, por lo que las relaciones que guardan entre ellas no suponen un criterio de peso en la elección de la base de datos.

Una base de datos NoSQL aporta por tanto la inmediatez que requiere una aplicación [en tiempo real con un acceso inmediato a los datos de recuento](#) que en una estructura SQL supondría un esfuerzo de procesamiento enorme.

La elección de [Firestore](#) de Google Cloud para implementar esta BD atiende a su mayor eficiencia en las consultas por trabajar con una estructura de documentos (así es como nombra a las estructuras de datos sencillas, por ejemplo, una carta) y colecciones (de documentos, por ejemplo, un mazo será una colección de documentos tipo carta).

De este modo, a diferencia de otras BBDD NoSQL convencionales, Firestore no nos devuelve un JSON con cada consulta, sino que nos devuelve únicamente el documento o la colección de documentos que nosotros solicitemos de manera selectiva. Una vez trabajamos en local con un documento o colección, cada nueva consulta sobre ese documento o colección sólo descargará los campos concretos que hayan cambiado de valor desde la última consulta y que han de ser refrescados. Como resultado, [el volumen de transaccionalidad de datos con el servidor será muy inferior](#), algo fundamental en una aplicación concebida para la actualización continua en tiempo real.

Por otra parte, los motivos para la elección de un formato SPA con Angular son fundamentalmente:

- Aporta versatilidad a la hora de acceder desde cualquier tipo de dispositivo a la aplicación haciéndola multiplataforma.
- Abre la puerta a su adaptación para soporte móvil con tecnologías como IONIC en un futuro.
- Desde el punto de vista académico, se utiliza la realización de este proyecto como medio de aprendizaje de una manera más profunda de TypeScript, un lenguaje de programación de muy amplia demanda en el mercado laboral pero que es abordado de manera menos intensa que otros como Java a lo largo del ciclo.
- Del mismo modo, sirve como pretexto para afianzar conocimientos básicos de herramientas de uso común en el diseño web con el uso de Bootstrap para todo el estilismo de la aplicación. Además, aporta uniformidad al aspecto visual de la UI que busca ser lo más sencilla posible con orientación a ser adaptada para el uso en dispositivos móviles que cuenten con un tamaño de pantalla reducido.
- La integración de la BD Firestore cuenta con una buena integración con JavaScript/TypeScript y Angular, resultando un tándem perfecto para desarrollar una app de juego en tiempo real donde el usuario debe ver las actualizaciones tanto propias como del resto de participantes del juego de manera instantánea, por lo que cualquier modificación en un registro de la BBDD debe ser refrescada de manera automática en los dispositivos de todos los jugadores conectados.

2.2. ESTUDIO DE LA SITUACIÓN ACTUAL

El estudio de idiomas extranjeros tiene un público cada vez más amplio, y su aprendizaje mediante plataformas digitales y aplicaciones (especialmente móviles) es una de las tendencias más en alza.

La necesidad de manejar una o varias lenguas extranjeras es cada vez mayor tanto a nivel académico como profesional, especialmente del inglés. Es requisito imprescindible para acceder a muchas oportunidades formativas o laborales, y en especial en las franjas de edad más jóvenes. Es este público además el que más uso hace de las nuevas tecnologías y de todo tipo de plataformas y aplicaciones (como la presente) para fines de cualquier índole.

2.3. EVALUACIÓN DE LAS ALTERNATIVAS

Existen en el mercado múltiples aplicaciones y plataformas para el estudio de idiomas extranjeros. Muchos de ellos con un alcance mucho más amplio que el de una mera mecánica de pregunta-respuesta (por ejemplo, la aplicación oficial de

Oxford incluye consultas a sus diccionarios), otros con una filosofía muy similar a la propuesta en este juego: preguntas de respuesta múltiple, de respuesta escrita, de habilidad auditiva e incluso de pronunciación. En la mayoría de los casos son las propias aplicaciones las que incluyen el 'material' de aprendizaje (vocabulario, temáticas, etc), esto ahorra esfuerzo al usuario pero suele dar muy pocas opciones de personalización del aprendizaje.

Para subsanar este inconveniente, QuizCard permite **configurar de manera total el set de preguntas y respuestas**. Para no perder comodidad y rapidez, permite la importación tanto de mazos completos de cartas como de cartas individuales pre-existentes de otros usuarios a nuestras propias colecciones.

Muchas de las aplicaciones actuales se centran además en la práctica individual del idioma, mientras otras permiten jugar con usuarios de cualquier parte del mundo al azar. La idea de QuizCard es ofrecer una plataforma desde la que poder practicar un idioma de manera conjunta entre usuarios conocidos, socializando el aprendizaje entre los grupos de amigos, familiares, compañeros de estudios o de trabajo, ... **Haciendo del estudio un actividad lúdica y social.**

2.4. ASPECTOS TEÓRICOS

El resumen de tecnologías utilizadas en el desarrollo de este proyecto es el siguiente:

- **TypeScript**: Lenguaje de programación para desarrollo web.
- **Angular**: Framework de desarrollo web.
- **Bootstrap**: Herramienta para diseño web.
- **Firestore**: Base de datos NoSQL de Google Cloud Platform basada en una estructura de documentos y colecciones.
- **VisualStudio Code**: IDE de desarrollo.

Capítulo 3 . ANÁLISIS

3.1. REQUISITOS

3.1.1.REQUISITOS NO FUNCIONALES

Número	Requerimiento	Descripción	Prioridad
RNF1	Inmediatez	Debe ofrecer al usuario la información relativa al mazo/carta actualizada en todo momento	5
RNF2	Eficiencia consumo de datos	La actualización de los cambios en la BD en tiempo real debe consumir el mínimo volumen de datos (MB) posible	5
RNF3	Facilidad de aprendizaje	Debe ser lo más sencilla e intuitiva, su usabilidad debe alejarse lo máximo posible de ser engorrosa o compleja	4
RNF4	Comodidad de uso	Debe ofrecer rapidez al usuario para: iniciar el juego, cambiar la modalidad de juego, consultar estadísticas	4
RNF5	Comodidad de acceso a material	Debe ofrecer una forma sencilla y rápida de generar material (cartas) para el estudio	4
RNF6	Multiplataforma	Debe ofrecer versatilidad al usuario a la hora de acceder desde distintos tipos de plataforma (móvil, PC, tablet, ...), tanto a nivel técnico como de diseño UI	3
RNF7	Adaptabilidad móvil	Es deseable que ofrezca facilidad para una posible futura adaptación para plataformas móviles	2
RNF8	Integridad de la información	No se manipula información sensible ni se trata de una aplicación cuyo uso afecte en ámbitos de importancia, un error en la integridad de los datos no es deseable pero no tendría consecuencias negativas	2
RNF9	Seguridad	La aplicación no maneja datos o información que deba ser protegida, por tanto no deberá contar con medidas de seguridad especiales	2

3.1.2.REQUISITOS FUNCIONALES

Número	Requerimiento	Descripción	Prioridad
RF1	El usuario podrá crear su propio material	El usuario debe poder crear sus propios mazos, configurando los idiomas utilizados, y las cartas que los compongan	5
RF2	El contenido creado por los usuarios será de acceso universal	Cualquier usuario tendrá acceso a todos los mazos y cartas creados en la aplicación, pudiendo importar a sus propios mazos las cartas que desee para agilizar la creación de material de estudio	5
RF3	Existirán distintas modalidades de juego	El usuario podrá escoger entre distintos modos de juego/estudio	4
RF4	Los resultados de los usuarios se almacenarán en estadísticas de juego	Se almacenarán todos los resultados de respuestas totales y aciertos de los usuarios tanto para las cartas individuales como para los mazos	5
RF5	Los usuarios de un grupo podrán ver las estadísticas de todos los miembros de ese grupo	Un usuario que juegue dentro de un grupo debe poder ver en tiempo real las estadísticas de todos los miembros de ese grupo en forma de ranking	4
RF6	Los usuarios deben poder ser añadidos a un grupo	Un usuario debe poder invitar a otros usuarios a jugar en un grupo al que él pertenezca	5

3.2. ESPECIFICACIÓN CASOS DE USO

Al tratarse de una app cuyos requerimientos funcionales a nivel de back-end son extremadamente sencillos y cuyo eje central es la experiencia de usuario, se encontró más adecuado centrar esta sección en los distintos casos de uso de la lógica front-end. Para ello se representan los distintos flujos de uso que puede seguir el usuario a través de las distintas secciones de la app.

Nombre	C01. Login
Descripción	El usuario inicia sesión en la aplicación
Precondición	Ninguna
Secuencia principal	<ol style="list-style-type: none">Opción 1: El usuario se logea si ya tiene una cuenta registrada (arriba izq.) Opción 2: El usuario se registra si aún no tiene una cuenta (arriba dcha.) Opción 3: El usuario ya se había logeado con anterioridad y su ID se carga por defecto (abajo)El usuario confirma que desea acceder con la cuenta que se muestra en pantalla (abajo)
Errores / Alternativas	En cualquiera de las pantallas el usuario cuenta con la posibilidad de cambiar a las otras opciones (elementos en rosa)
Postcondición	El usuario accede a la pantalla principal de la aplicación
Notas	La lógica back-end de registro y logeo de usuarios excede el alcance de este proyecto

Nombre	C02. Página principal
Descripción	El usuario selecciona una de las 4 opciones de la página de inicio de la aplicación
Precondición	Estar logeado
Secuencia principal	<ol style="list-style-type: none">Mostrar las 4 opciones principales de la app: jugar, ver sus mazos, ver sus estadísticas y acceder al repositorio público de la aplicación
Errores / Alternativas	Ninguno
Postcondición	El usuario accede a la pantalla principal de la opción elegida
Notas	Ninguna

Nombre	C03. Sección 'My Statistics'
Descripción	El usuario ve las estadísticas de todos sus mazos
Precondición	Seleccionar la opción 'My Statistics' en la página principal de la app
Secuencia principal	<ol style="list-style-type: none"> 1. Mostrar todos los mazos del usuario acompañados de una barra indicadora de su porcentaje de aciertos en cada mazo. La barra será de color verde si supera el 50% de aciertos de roja si es inferior 2. Cuando el usuario pase el ratón por encima podrá ver los detalles (totales numéricos) de sus estadísticas
Errores / Alternativas	Ninguno
Postcondición	El usuario visualiza una interfaz que muestra las estadísticas de cada uno de sus mazos
Notas	Final de flujo UI. La única alternativa es permanecer en esta pantalla o volver a pantallas anteriores

Nombre	C04. Sección 'Repository'
Descripción	El usuario accede al repositorio público de la aplicación
Precondición	Seleccionar la opción 'Repository' en la página principal de la app
Secuencia principal	<ol style="list-style-type: none"> 1. Mostrar dos desplegables para seleccionar los idiomas por los que se quiere filtrar la búsqueda de mazos 2. Si se efectúa la búsqueda: Mostrar todos los mazos de los lenguajes seleccionados en la búsqueda junto a su número de cartas 3. Si se selecciona un mazo: Mostrar todas las cartas del mazo seleccionado 4. Si se selecciona una o varias cartas: Activar el botón 'Copy' 5. Si se pulsa 'Copy': Mostrar el listado de mazos del usuario para que seleccione uno al que volcar las cartas seleccionadas 6. Si se selecciona un mazo propio: Crear nuevas cartas en el mazo del usuario con la pregunta-respuesta de las seleccionadas y las estadísticas (a cero) para todos los jugadores del mazo
Errores / Alternativas	Ninguno
Postcondición	El usuario visualiza la información pertinente según el estado de actividad en que se encuentre, de acuerdo a lo descrito en la secuencia principal
Notas	Habrà un lenguaje por defecto preconfigurado para ambas opciones de búsqueda

Nombre	C05. Sección 'My Decks'
Descripción	El usuario ve todos sus mazos
Precondición	Seleccionar la opción 'My Decks' en la página principal de la app
Secuencia principal	<ol style="list-style-type: none"> 1. Mostrar todos los mazos del usuario junto a su número de cartas y sus estadísticas 2. Si se pulsa el botón 'Add': Mostrar los inputs necesarios para añadir un nuevo mazo 3. Si se clicka sobre un mazo: Activar botones 'Edit' y 'Remove'. <ol style="list-style-type: none"> 3.1. Si se pulsa 'Edit': Dar la opción de cambiar el nombre, mostrar los botones 'Update' y 'Cancel' <ol style="list-style-type: none"> 3.1.1. Si se pulsa 'Update' confirmar los cambios 3.1.2. Si se pulsa 'Cancel' descartas las modificaciones 3.2. Si se pulsa 'Remove': Eliminar al usuario de dicho mazo 4. Si se pulsa el botón 'Open' de un mazo: Mostrar todas las cartas de ese mazo junto a las estadísticas del usuario para cada uno de ellos, y la opción de añadir a un usuario al mazo ('Invite a friend!') <ol style="list-style-type: none"> 4.1. Si se clicka sobre una carta: Activar botones 'Edit' y 'Remove'. <ol style="list-style-type: none"> 4.1.1. Si se pulsa 'Edit': Dar la opción de cambiar la pregunta y la respuesta, mostrar los botones 'Update' y 'Cancel' <ol style="list-style-type: none"> 4.1.1.1 Si se pulsa 'Update' confirmar los cambios 4.1.1.2 Si se pulsa 'Cancel' descartas las modificaciones 4.1.2. Si se pulsa 'Remove': Eliminar la carta de dicho mazo 4.2. Si se pulsa 'Invite a friend!' mostrar un input para introducir el email del usuario a invitar a unirse al mazo, mostrar los botones 'Invite' y 'Cancel'. <ol style="list-style-type: none"> 4.2.1. Si se pulsa 'Invite': Dado que el back-end de gestión de usuarios no está implementada, no cuenta con funcionalidad implementada) 4.2.2. Si se pulsa 'Cancel' descartas las modificaciones
Errores / Alternativas	Que no se seleccione alguno de los idiomas (o ambos) al crear un mazo, se controla configurando dos lenguajes por defecto
Postcondición	El usuario visualiza la información pertinente según el estado de actividad en que se encuentre, de acuerdo a lo descrito en la secuencia principal
Notas	<ul style="list-style-type: none"> - Los botones 'Add', 'Edit' 'Update', 'Remove' y 'Cancel' serán visibles o no y aparecerán como activados o no en función del estado de actividad en que se encuentre el usuario. - Habrá lenguajes por defecto preconfigurados al crear un mazo nuevo. - Cuando un usuario borra un mazo ('Remove') NO se elimina el mazo de la BD a menos que este usuario sea el único que juega a dicho mazo. De otro modo, se elimina al usuario de las estadísticas de ese mazo y de sus cartas, pero se conserva el mazo con sus otros participantes.

Nombre	C06. Sección 'Play'
Descripción	El usuario selecciona una de las 4 modalidades de juego de la aplicación
Precondición	Seleccionar la opción 'Play' en la página principal de la app
Secuencia principal	<ol style="list-style-type: none"> 1. Mostrar las 4 modalidades de juego de la app: 'Quiz', 'Writting', 'Listening', 'Training' 2. Si el usuario pulsa en una de las modalidades: Se muestra la lista de mazos del usuario para que elija a cuál jugar 3. Si el usuario elige un mazo: Acceder a la pantalla de juego de la modalidad seleccionada con el mazo seleccionado
Errores / Alternativas	Ninguno
Postcondición	El usuario accede a la pantalla de juego de la modalidad seleccionada con el mazo elegido
Notas	Ninguna

Nombre	C06.1. Modalidad de juego 'Quiz'
Descripción	El usuario juega en una modalidad de pregunta única con respuesta múltiple
Precondición	Seleccionar la opción 'Quiz' y un mazo al que jugar en la sección 'Play'
Secuencia principal	<ol style="list-style-type: none"> 1. El juego elige al azar cuatro cartas: una a la que jugar y las otras tres para configurar las respuestas múltiples (habrá 4 respuestas a elegir) 2. Se muestra la pregunta de la carta elegida para jugar y las respuestas de las cuatro cartas 3. El usuario pulsa sobre una de las respuestas <ol style="list-style-type: none"> 3.1. La respuesta correcta se mostrará en color verde siempre 3.2. Si la respuesta elegida es distinta a la correcta se mostrará en color rojo 4. Se actualizan las estadísticas de ese usuario con los resultados de la jugada <ol style="list-style-type: none"> 4.1. Se actualizan las estadísticas globales para ese mazo 4.2. Se actualizan las estadísticas de la carta elegida para jugar 5. Tras cada respuesta se genera automáticamente una nueva elección aleatoria de cartas (nueva jugada, punto 1) tras unos segundos de espera para que el jugador pueda ver la corrección del resultado
Errores / Alternativas	Ninguno
Postcondición	Ninguna
Notas	<ul style="list-style-type: none"> - Se mostrará en todo momento el ranking de todos los jugadores del mazo actualizado en tiempo real en la parte inferior de la pantalla - El usuario podrá invertir el orden de los idiomas (pregunta en idioma 1 y respuestas en idioma 2 o viceversa), en ese caso se elegirán de nuevo las cartas (punto 1) para evitar que la respuesta ya sea conocida

Nombre	C06.2. Modalidad de juego 'Writing'
Descripción	El usuario juega en una modalidad de pregunta leída con respuesta escrita
Precondición	Seleccionar la opción 'Writing' y un mazo al que jugar en la sección 'Play'
Secuencia principal	<ol style="list-style-type: none"> 1. El juego elige al azar una carta 2. Se muestra la pregunta de la carta, un input para que el usuario escriba su respuesta y un botón para que el usuario pueda confirmar su respuesta ('Check') 3. El usuario pulsa 'Check' <ol style="list-style-type: none"> 3.1. La respuesta correcta se mostrará bajo el input de respuesta 3.2. Si la respuesta escrita es correcta se mostrará en color verde, de lo contrario se mostrará en color rojo 3.3. El botón 'Check' se convertirá en 'Next' 4. Se actualizan las estadísticas de ese usuario con los resultados de la jugada <ol style="list-style-type: none"> 4.1. Se actualizan las estadísticas globales para ese mazo 4.2. Se actualizan las estadísticas de la carta elegida para jugar 5. Si el usuario pulsa 'Next' se genera una nueva elección aleatoria de carta (nueva jugada, punto 1)
Errores / Alternativas	Ninguno
Postcondición	Ninguna
Notas	<ul style="list-style-type: none"> - Se mostrará en todo momento el ranking de todos los jugadores del mazo actualizado en tiempo real en la parte inferior de la pantalla - El usuario podrá invertir el orden de los idiomas (pregunta en idioma 1 y respuestas en idioma 2 o viceversa), en ese caso se elegirán de nuevo las cartas (punto 1) para evitar que la respuesta ya sea conocida

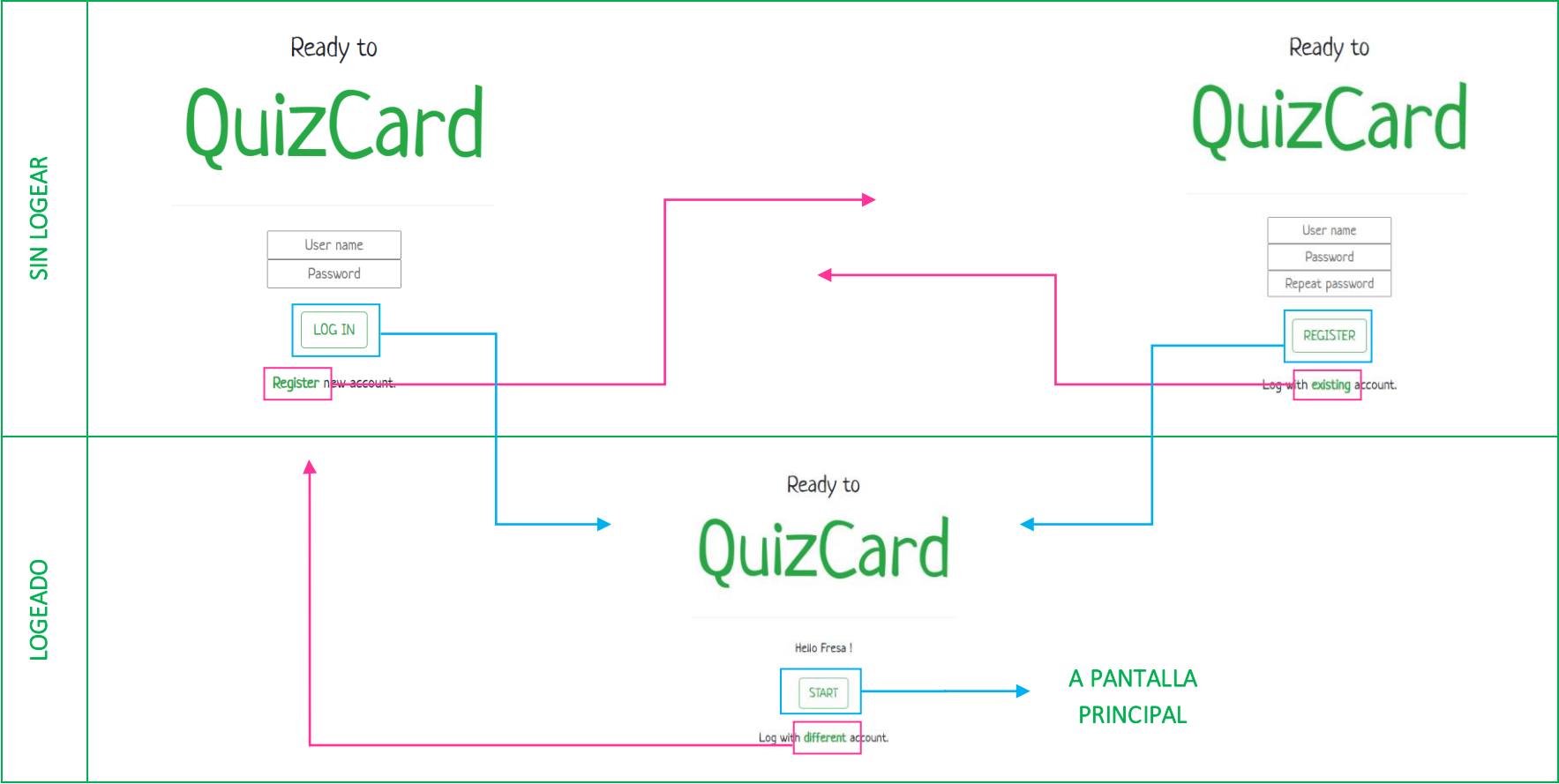
Nombre	C06.3. Modalidad de juego 'Listening'
Descripción	El usuario juega en una modalidad de pregunta auditiva con respuesta escrita
Precondición	Seleccionar la opción 'Listening' y un mazo al que jugar en la sección 'Play'
Secuencia principal	<ol style="list-style-type: none"> 1. El juego elige al azar una carta 2. Se muestra la pregunta de la carta, un input para que el usuario escriba su respuesta, un botón para que el usuario pueda confirmar su respuesta ('Check'), y tres botones para que el usuario escuche la pregunta ('Slow', 'Medium', 'Fast') 3. El usuario pulsa 'Check' <ol style="list-style-type: none"> 3.1. La respuesta correcta se mostrará bajo el input de respuesta 3.2. Si la respuesta escrita es correcta se mostrará en color verde, de lo contrario se mostrará en color rojo 3.3. El botón 'Check' se convertirá en 'Next' 4. Se actualizan las estadísticas de ese usuario con los resultados de la jugada <ol style="list-style-type: none"> 4.1. Se actualizan las estadísticas globales para ese mazo 4.2. Se actualizan las estadísticas de la carta elegida para jugar 5. Si el usuario pulsa 'Next' se genera una nueva elección aleatoria de carta (nueva jugada, punto 1)
Errores / Alternativas	Ninguno
Postcondición	<p>El usuario accede a la pantalla de juego de la modalidad seleccionada con el mazo elegido</p> <p>Tras cada respuesta se genera una nueva elección aleatoria de cartas (nueva jugada)</p>
Notas	<ul style="list-style-type: none"> - Se mostrará en todo momento el ranking de todos los jugadores del mazo actualizado en tiempo real en la parte inferior de la pantalla - El usuario podrá invertir el orden de los idiomas (pregunta en idioma 1 y respuestas en idioma 2 o viceversa), en ese caso se elegirán de nuevo las cartas (punto 1) para evitar que la respuesta ya sea conocida - El usuario podrá escuchar la pregunta tantas veces como desee escogiendo además la velocidad en que es reproducida ('Slow', 'Medium', 'Fast')

Nombre	C06.4. Modalidad de juego 'Training'
Descripción	El usuario juega en una modalidad de pregunta única con respuesta escrita donde los resultados obtenidos no se registrarán en sus estadísticas
Precondición	Seleccionar la opción 'Training' y un mazo al que jugar en la sección 'Play'
Secuencia principal	<ol style="list-style-type: none"> 1. El juego elige al azar una carta 2. Se muestra la pregunta de la carta, un botón para visualizar la respuesta correcta ('Check'), y tres botones para que el usuario escuche la pregunta ('Slow', 'Medium', 'Fast') 3. El usuario pulsa 'Check' <ol style="list-style-type: none"> 3.1. Se mostrará la respuesta correcta 3.2. El botón 'Check' se convertirá en 'Next' 4. Si el usuario pulsa 'Next' se genera una nueva elección aleatoria de carta (nueva jugada, punto 1)
Errores / Alternativas	Ninguno
Postcondición	Ninguna
Notas	<ul style="list-style-type: none"> - Se mostrará en todo momento el ranking de todos los jugadores del mazo actualizado en tiempo real en la parte inferior de la pantalla - El usuario podrá invertir el orden de los idiomas (pregunta en idioma 1 y respuestas en idioma 2 o viceversa) - En esta modalidad no se actualizarán las estadísticas del usuario con los resultados de las jugadas

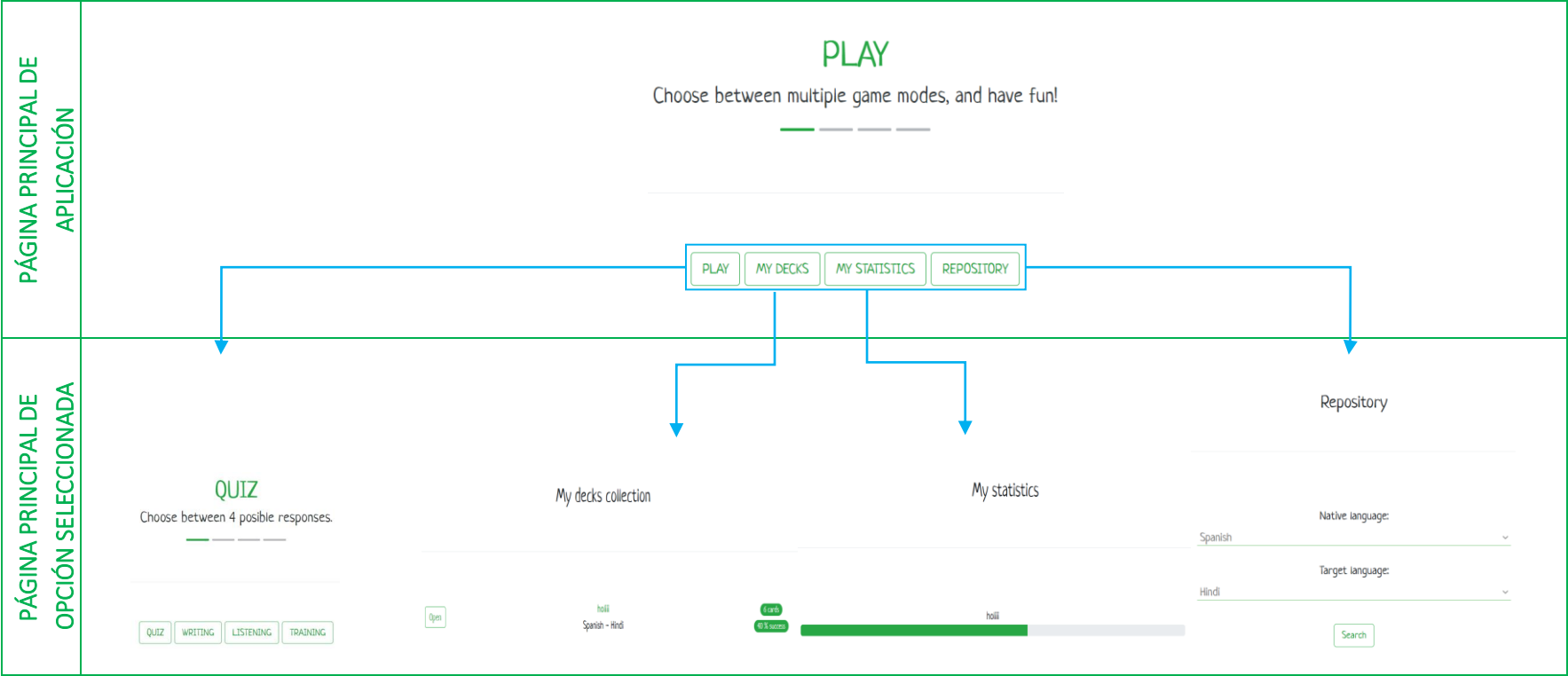
Nombre	C07. NavBar
Descripción	El usuario podrá hacer logout, navegar hacia atrás o navegar a cualquier sección concreta de la app
Precondición	Estar logeado y haber accedido a la aplicación
Secuencia principal	<ol style="list-style-type: none"> 1. Si el usuario pulsa '< Quizcard': Navegar hacia atrás 2. Si el usuario pulsa '%UserName%': Mostrar opciones de navegación <ol style="list-style-type: none"> 2.1. Si el usuario pulsa 'Home': Navegar a página principal 2.2. Si el usuario pulsa 'Play': Navegar a página 'Play' 2.3. Si el usuario pulsa 'My Decks': Navegar a página 'My Decks' 2.4. Si el usuario pulsa 'Statistics': Navegar a página 'My Statistics' 2.5. Si el usuario pulsa 'Repository': Navegar a página 'Repository' 2.6. Si el usuario pulsa 'Sign off': Deslogearse de la aplicación
Errores / Alternativas	Ninguno
Postcondición	El usuario accede a la pantalla elegida. En caso de deslogearse, accede a la página de login
Notas	Las opciones de navegación aparecerán por defecto ocultas, sólo se mostrarán cuando el usuario pulse el botón superior derecho (este mostrará su nombre)

A lo largo de toda la aplicación, para gestionar los componentes que han de cargarse en cada situación se utilizan los parámetros de la URL (UserID, DeckID, cardID, ...). Dado que son los únicos parámetros que se manejan, no fue imprescindible emplear otros medios para comunicar información entre componentes, pues el resto de datos mostrados en pantalla se obtienen en tiempo de ejecución mediante observables.

C01. LOGEARSE EN LA APLICACIÓN



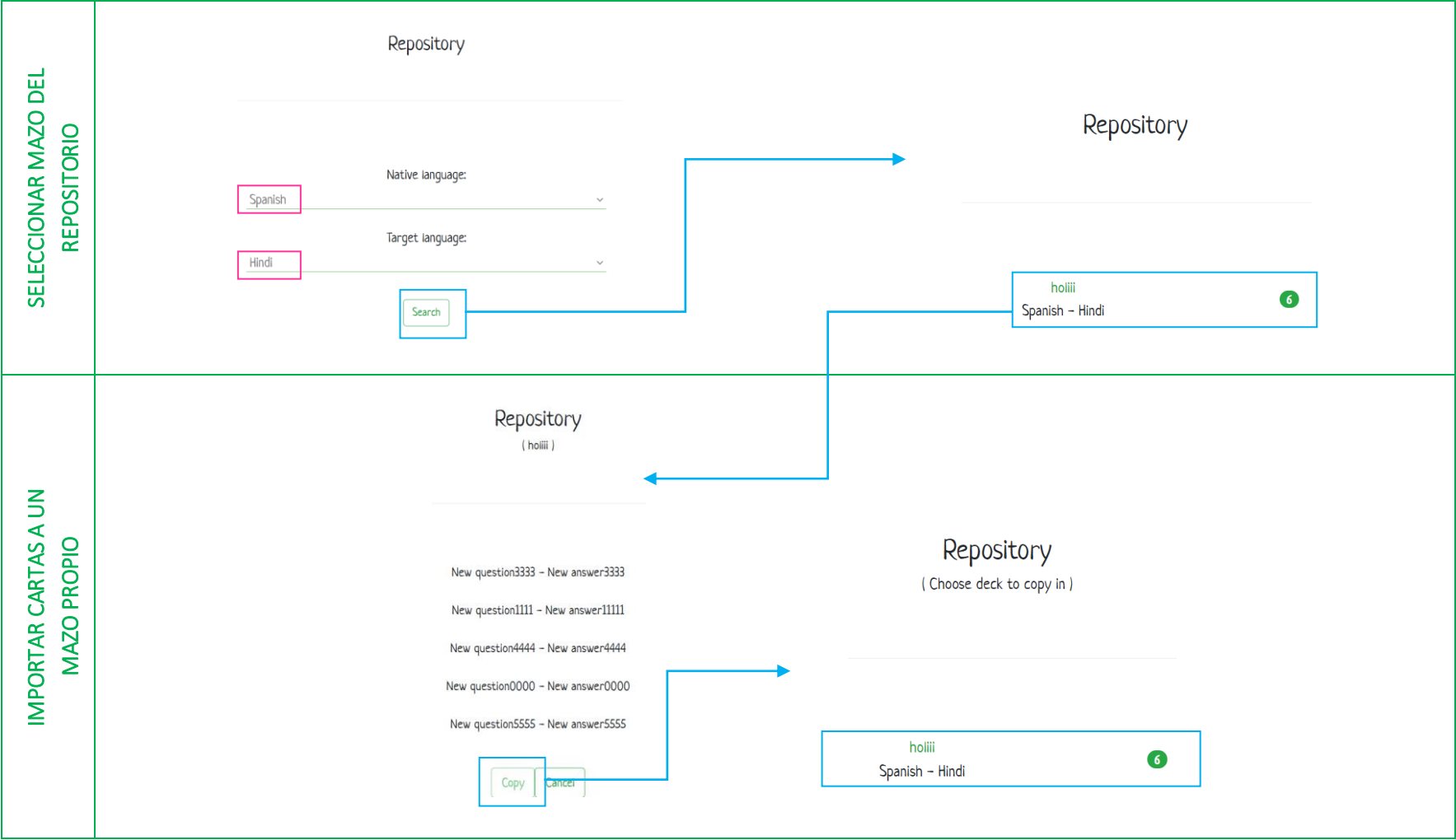
C02. SELECCIONAR OPCIÓN EN PÁGINA PRINCIPAL



C03. SECCIÓN 'MY STATISTICS'

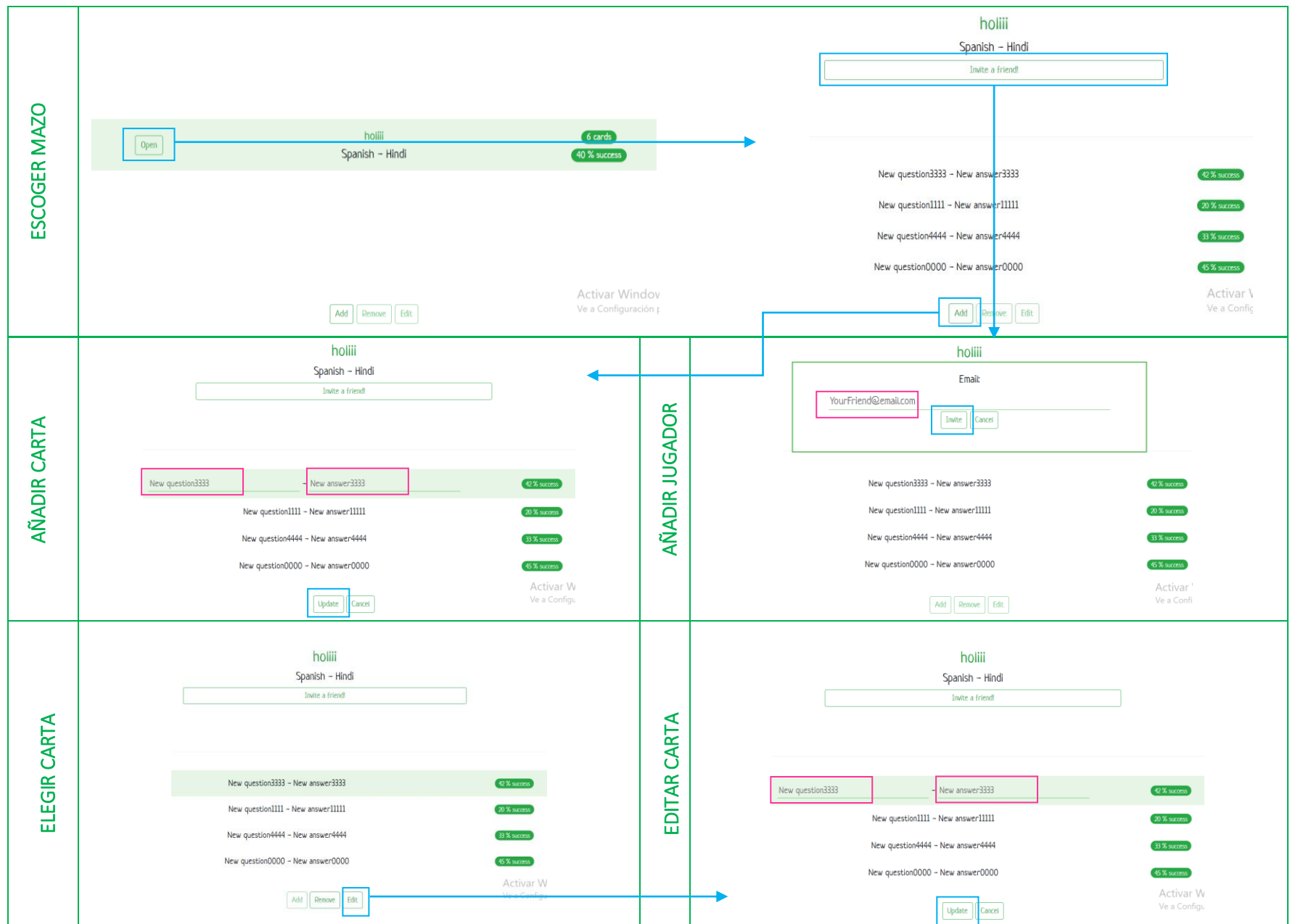
MAYORÍA DE ACIERTOS	MAYORÍA DE FALLOS
<div><div>My statistics</div><div><div></div></div></div> <div><div>My statistics</div><div><div>Total: 61</div><div>Success: 36</div><div>59%</div></div></div>	<div><div>My statistics</div><div><div></div></div></div> <div><div>My statistics</div><div><div>Total: 42</div><div>Success: 17</div><div>40%</div></div></div>

C04. SECCIÓN 'REPOSITORY'



C05. SECCIÓN 'MY DECKS'

AÑADIR MAZO	<div><div>My decks collection</div><div><div>Open</div><div>holili Spanish - Hindi</div><div>6 cards 40 % success</div></div><div><div>Add</div><div>Remove</div><div>Edit</div></div><div>Activar W Ve a Configu</div></div> <div><div>My decks collection</div><div><div>Open</div><div><div><div>My new deck !</div><div>Spanish</div><div>English</div></div><div><div>Name:</div><div>Native language:</div><div>Target language:</div></div></div><div>6 cards 40 % success</div></div><div><div>Add</div><div>Cancel</div></div><div>Activar V Ve a Config</div></div>
EDITAR MAZO	<div><div>Open</div><div>holili Spanish - Hindi</div><div>6 cards 40 % success</div></div> <div><div>Add</div><div>Remove</div><div>Edit</div></div> <div>Activar Window Ve a Configuración</div> <div><div>Open</div><div>holili Spanish - Hindi</div><div>6 cards 40 % success</div></div> <div><div>Update</div><div>Cancel</div></div> <div>Activar Window Ve a Configuración</div>
BORRAR MAZO	<div><div>Open</div><div>holili Spanish - Hindi</div><div>6 cards 40 % success</div></div> <div><div>Add</div><div>Remove</div><div>Edit</div></div> <div>Activar Window Ve a Configuración</div>



BORRAR CARTA

holiii

Spanish - Hindi

Invite a friend!

New question3333 - New answer3333

45 % success

New question1111 - New answer1111

30 % success

New question4444 - New answer4444

33 % success

New question0000 - New answer0000

45 % success

Add

Remove

Edit

Activar W

Ve a Configu

C06. SECCIÓN 'PLAY'

ELEGIR MODALIDAD

QUIZ

Choose between 4 posible responses.

QUIZ

WRITING

LISTENING

TRAINING

ELEGIR MAZO

Choose a deck to play

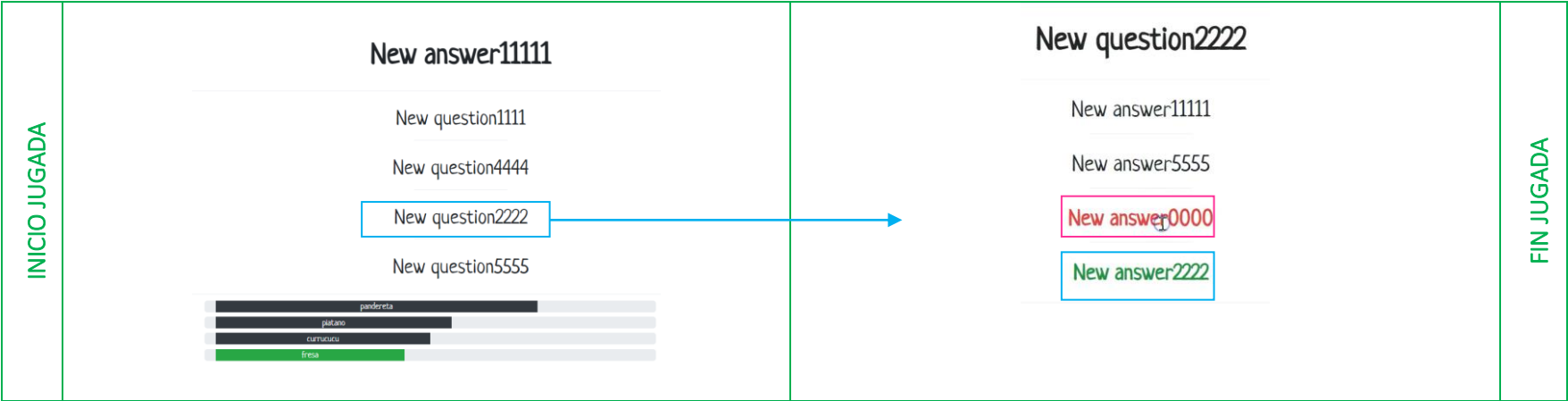
holiii

Spanish - Hindi

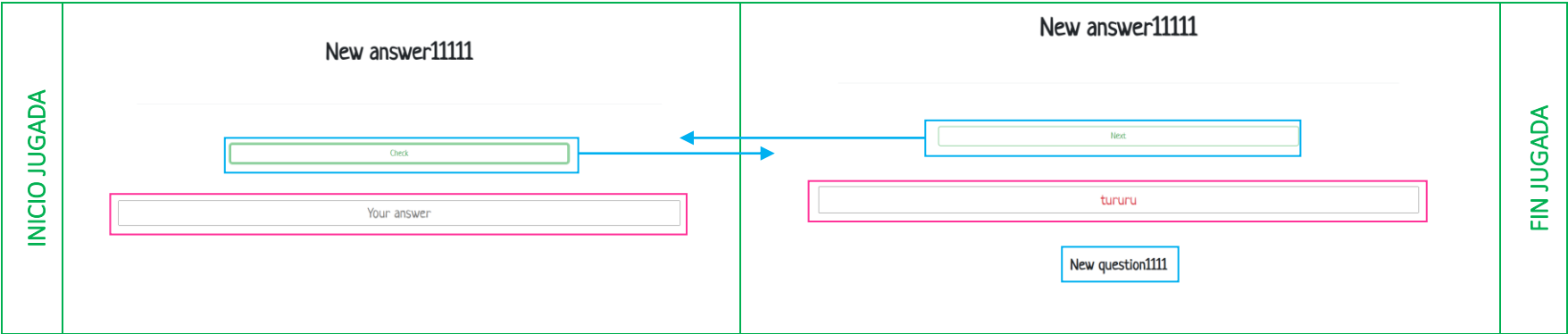
→

A PANTALLA DE JUEGO

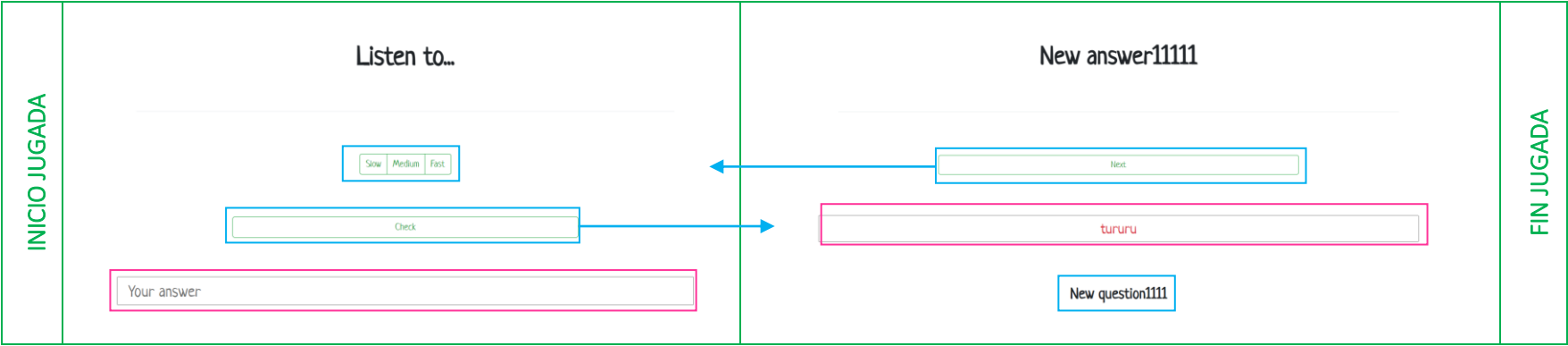
C06.1 SECCIÓN ‘QUIZ’



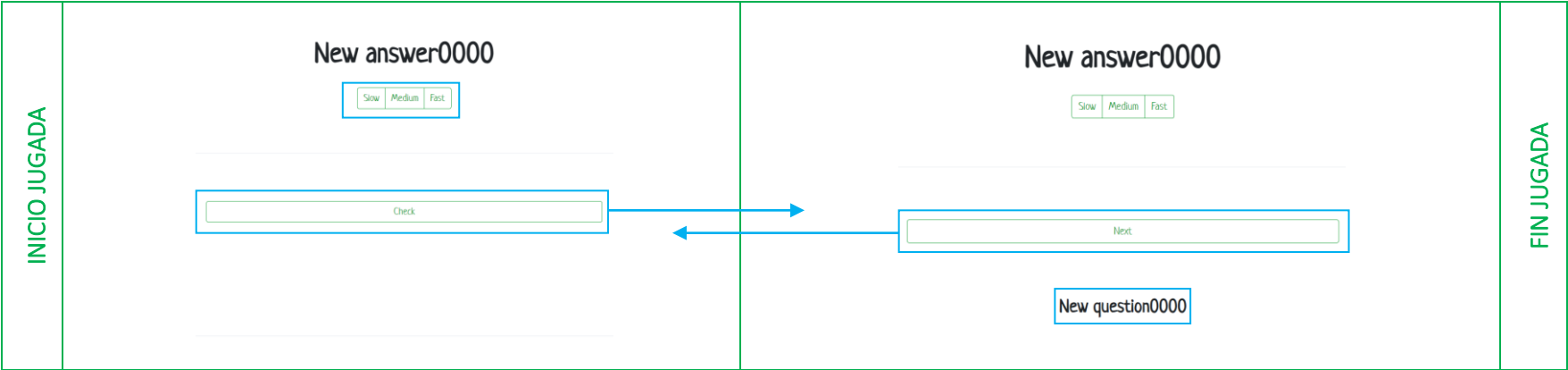
C06.2 SECCIÓN ‘WRITING’




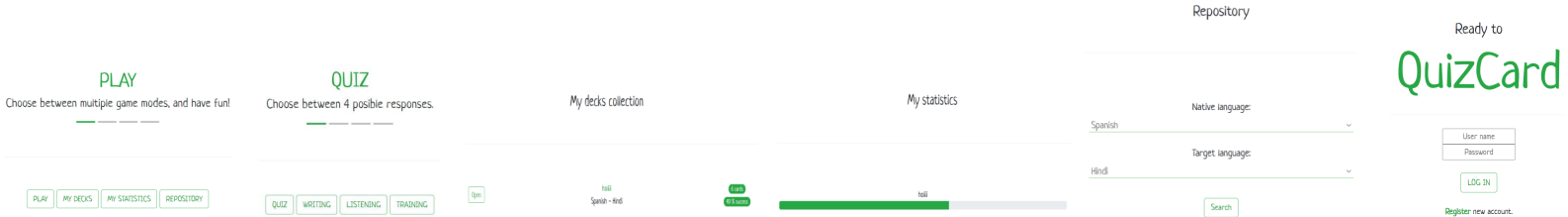
C06.3 SECCIÓN 'LISTENING'



C06.4 SECCIÓN 'TRAINING'



C07. NAVBAR

VISUALIZAR NAVBAR	
NAVEGAR A SECCIÓN	

Capítulo 4 . DISEÑO DEL SISTEMA

4.1. ARQUITECTURA DEL SISTEMA

4.1.1. DIAGRAMA DE PAQUETES

Diagrama **general** de la aplicación:

- Components : Contiene los componentes de la UI
- Pojos : Contiene los modelos de datos con que trabaja la aplicación
- Services : Contiene los servicios de comunicación con la BD

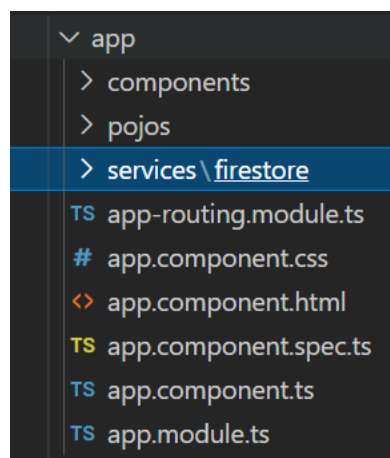
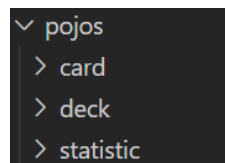


Diagrama del directorio '**Pojos**':

- Cada uno de los subdirectorios contiene el directorio TS con la implementación del modelo de datos concreto y la lógica propia de dicha estructura de datos.



Ejemplo de estructura de ficheros de un pojo particular:

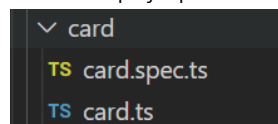
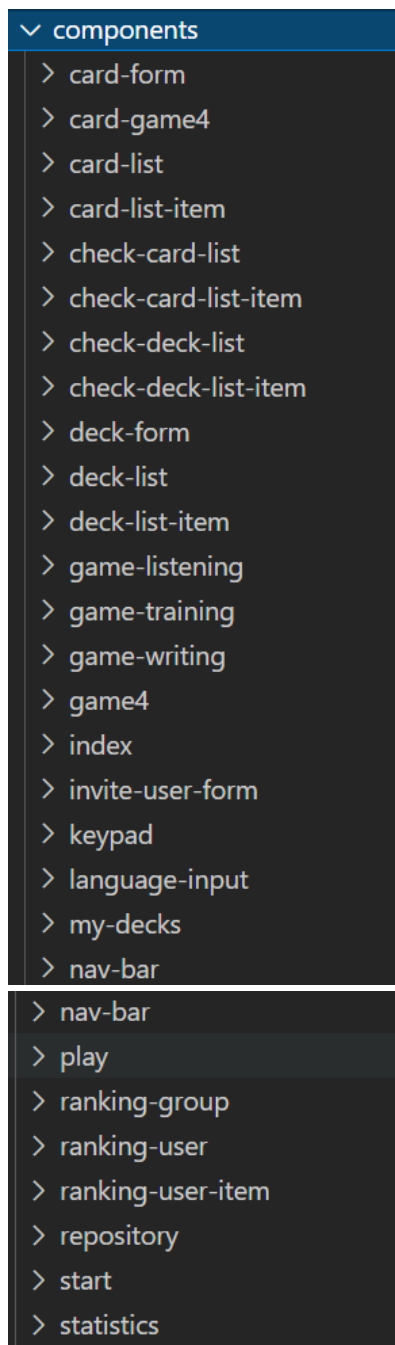


Diagrama del directorio 'Components':

- Cada uno de los subdirectorios contiene los ficheros HTML, CSS y TS del componente en cuestión



Ejemplo de la estructura de ficheros de un componente particular:

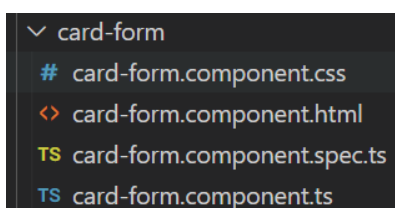


Diagrama del directorio 'Services':

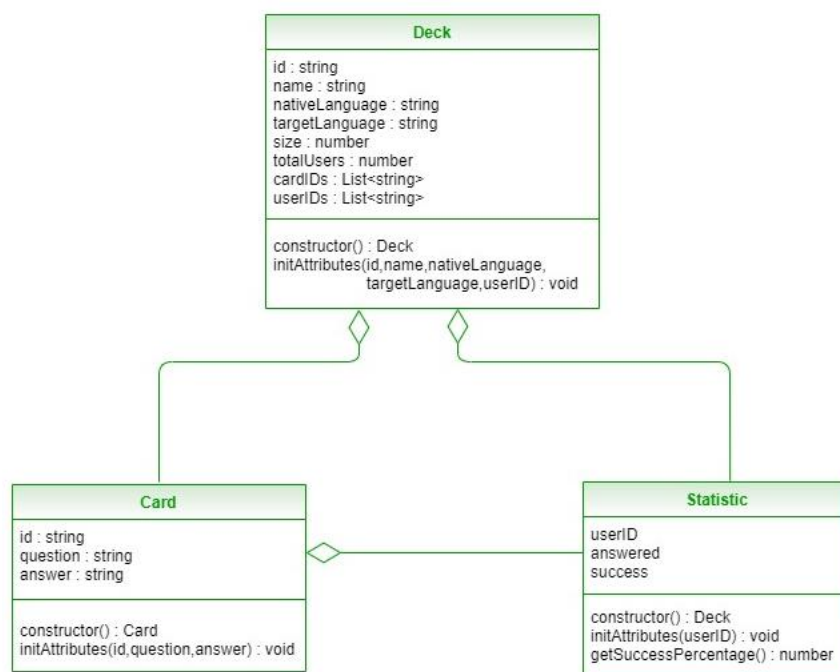
- Este directorio sólo contiene el subdirectorio 'Firestore' por lo que se muestra directamente su contenido: el archivo TS con la lógica implementada para realizar todas las operaciones sobre la BD del proyecto.

```
services \ firestore
TS firestore-decks.service.spec.ts
TS firestore-decks.service.ts
```

4.2. DISEÑO DE CLASES

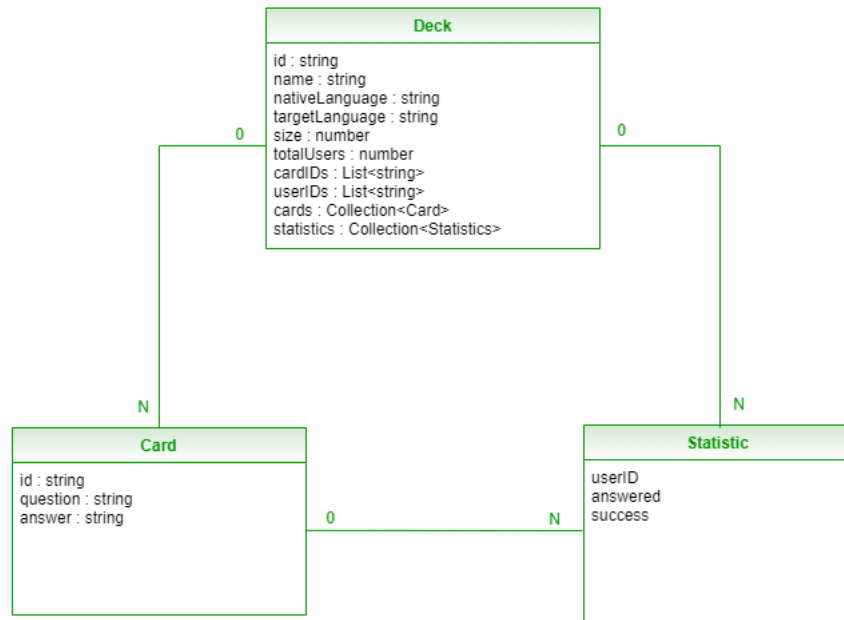
4.2.1. DIAGRAMA DE CLASES

Conceptualmente son agregaciones, en la implementación no porque los pojos sólo contienen listas de los ID de las instancias asociadas (trabajamos con observables, hacemos las peticiones de los objetos en tiempo real, no los precargamos)

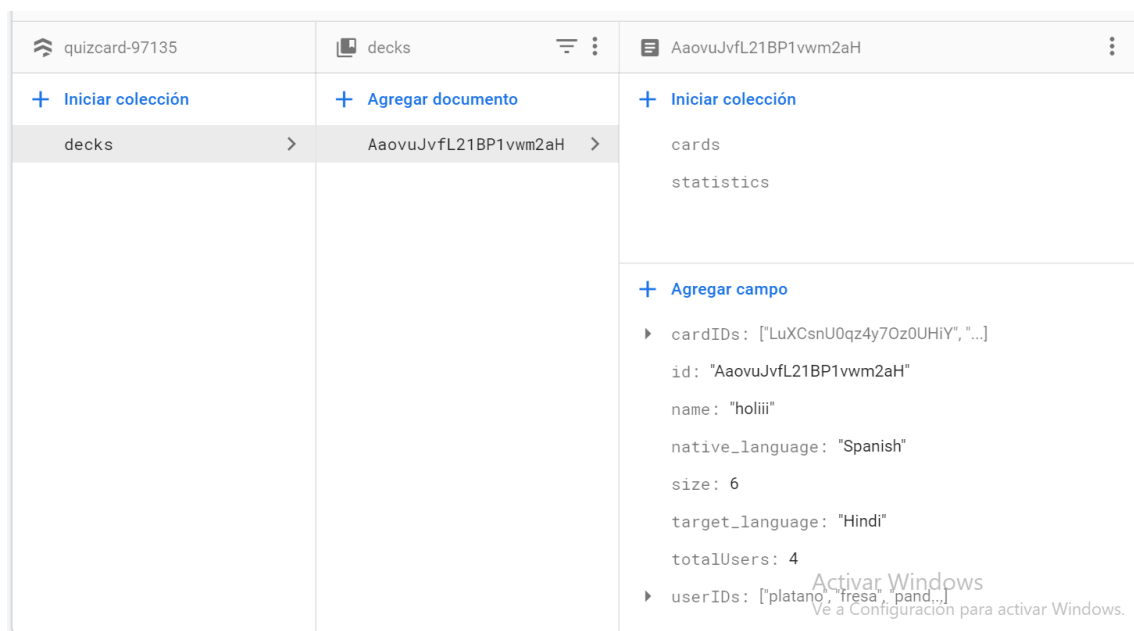


4.3. DISEÑO DE BASES DE DATOS

Hacer UML con sus *-0 etc como el de arriba pero cambiando los conectores y poniendo listas de objetos y listas de ID para acceso rápido



Capturas de pantalla de la BD en Google Cloud Console. Se observa la diferencia entre los campos sencillos y aquellos que constituyen colecciones.



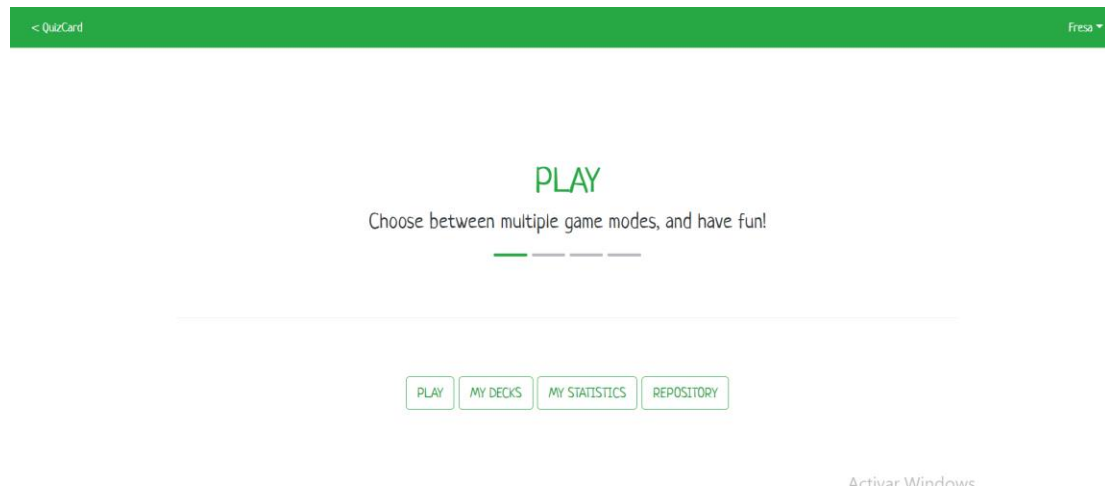
<div>AaovuJvfl21BP1vwm2aH</div> <div><div>+ Iniciar colección</div><div>cards</div><div>statistics</div></div> <div><div>+ Agregar campo</div><div>cardIDs: ["LuXCsnU0qz4y70; "...]</div><div>id: "AaovuJvfl21BP1vwm2aH"</div><div>name: "holiii"</div><div>native_language: "Spanish"</div><div>size: 6</div><div>target_language: "Hindi"</div><div>totalUsers: 4</div></div>	<div>cards</div> <div><div>+ Agregar documento</div><div>BXQTF4tSRaPSrX07xm4n</div><div>LuXCsnU0qz4y70z0UHiY</div><div>SvTpMEwMZc2jeEjtje9q</div><div>jnnNnJ23WxM5Mf98MWxx</div><div>IggQRix3RHw2CXMLsoMy</div><div>pTSqfxYg9Bs7DQGvjzfb</div></div>	<div>BXQTF4tSRaPSrX07xm4n</div> <div><div>+ Iniciar colección</div><div>statistics</div></div> <div><div>+ Agregar campo</div><div>answer: "New answer3333"</div><div>id: "BXQTF4tSRaPSrX07xm4n"</div><div>question: "New question3333"</div></div> <div>Activar Windows</div> <div>Ve a Configuración para activar Windows.</div>
---	--	--

<div>AaovuJvfl21BP1vwm2aH</div> <div><div>+ Iniciar colección</div><div>cards</div><div>statistics</div></div> <div><div>+ Agregar campo</div><div>cardIDs: ["LuXCsnU0qz4y70; "...]</div><div>id: "AaovuJvfl21BP1vwm2aH"</div><div>name: "holiii"</div><div>native_language: "Spanish"</div><div>size: 6</div><div>target_language: "Hindi"</div><div>totalUsers: 4</div></div>	<div>statistics</div> <div><div>+ Agregar documento</div><div>currucucu</div><div>fresa</div><div>pandereta</div><div>platano</div></div>	<div>currucucu</div> <div><div>+ Iniciar colección</div><div>+ Agregar campo</div><div>answered: 14</div><div>success: 7</div><div>userID: "currucucu"</div></div> <div>Activar Windows</div> <div>Ve a Configuración para activar Windows.</div>
---	---	---

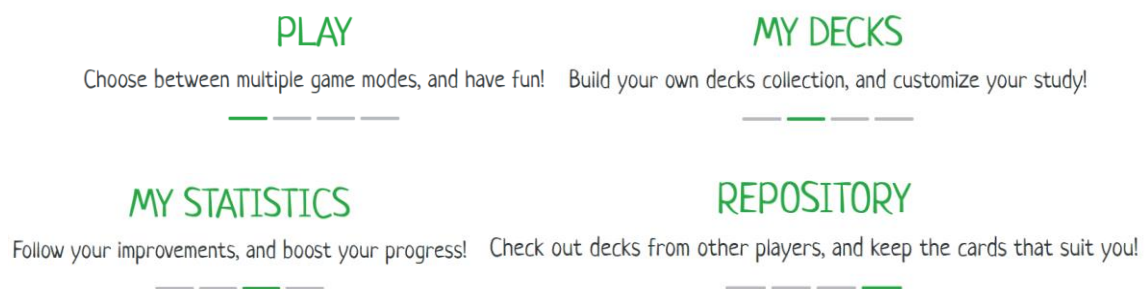
<div>BXQTF4tSRaPSrX07xm4n</div> <div><div>+ Iniciar colección</div><div>statistics</div></div> <div><div>+ Agregar campo</div><div>answer: "New answer3333"</div><div>id: "BXQTF4tSRaPSrX07xm4n"</div><div>question: "New question3333"</div></div>	<div>statistics</div> <div><div>+ Agregar documento</div><div>currucucu</div><div>fresa</div><div>pandereta</div><div>platano</div></div>	<div>currucucu</div> <div><div>+ Iniciar colección</div><div>+ Agregar campo</div><div>answered: 3</div><div>success: 1</div><div>userID: "currucucu"</div></div>
---	---	---

4.4. DISEÑO DE LA INTERFAZ

Página principal

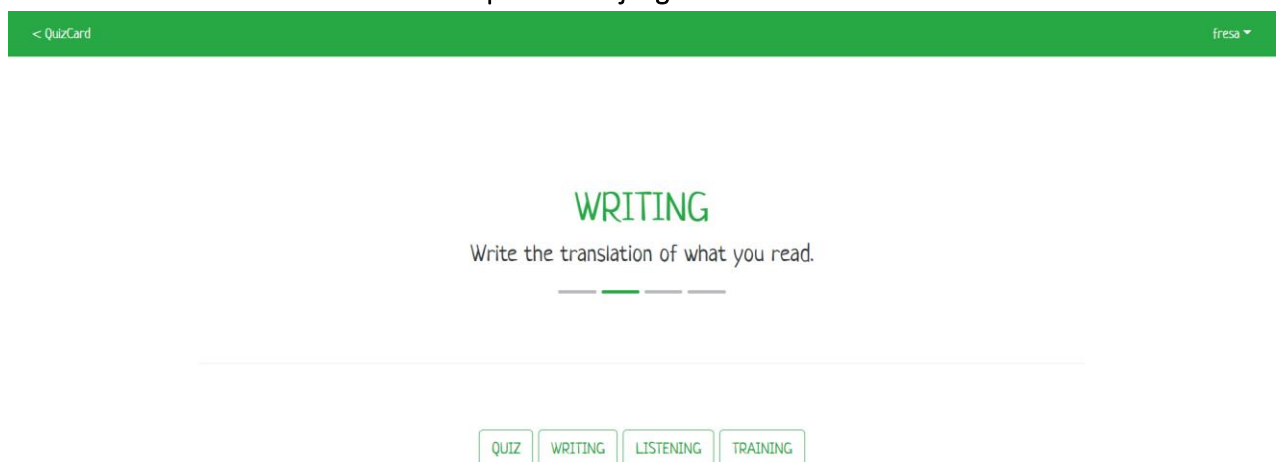


Carrousel de la página principal



Sección 'Play'

Opciones de juego



Carrousel de opciones de juego

QUIZ

Choose between 4 posible responses.

— — — —

TRAINING

Just give your vocabulary a review.

— — — —

WRITING

Write the translation of what you read.

— — — —

LISTENING

Write the translation of what you hear.

— — — —

Elegir mazo para jugar

< QuizCard

fresa ▾

Choose a deck to play

holiii
Spanish - Hindi

6

Modalidad 'Quiz'

(Inicio de jugada)

< QuizCard

Hindi -> Spanish

fresa ▾

New answer11111

New question1111

New question4444

New question2222

New question5555

	pandereta	
	patano	
	curruco	
	fresa	

(Fin de jugada)

New answer0000

New question5555

New question3333

New question0000

New question2222

pandereta	
patano	
cumucuu	
fresa	

Modalidad 'Writing'

(Inicio de jugada)

New answer1111

Check

Your answer

pandereta	
patano	
cumucuu	
fresa	

(Fin de jugada)

New answer1111

Next

tururu

New question1111

pandereta	
patano	
cumucuu	
fresa	

Modalidad 'Listening'

(Inicio de partida)

Listen to...

Slow Medium Fast

Check

Your answer



(Fin de partida)

Listen to...

Slow Medium Fast

Next

tururu

New answer2222



Activar Windows
Ve a Configuración para activar Windows.

Modalidad 'Training'

(Inicio de partida)

< QuizCard

Hindi ⇌ Spanish

fresa ▼

New answer0000

Slow

Medium

Fast

Check

pandereta

platano

curucucu

fresa

(Fin de partida)

< QuizCard

Hindi ⇌ Spanish

fresa ▼

New answer0000

Slow

Medium

Fast

Next

New question0000

pandereta

platano

curucucu

fresa

Sección ' My Decks'

< QuizCard

fresa ▼

My decks collection

Open

holiii
Spanish - Hindi

6 cards
100 % success

Add

Remove

Edit

Activar Windows

Ve a Configuración para activar Windows.

Añadir nuevo mazo

< QuizCard

fresa

My decks collection

Open

Name:

My new deck !

Native language:

Spanish

Target language:

English

6 cards

40 % success

Add

Cancel

Activar Windows
Ve a Configuración para activar Windows.

Mazo seleccionado / Hover

< QuizCard

fresa

My decks collection

Open

holiii

Spanish - Hindi

6 cards

40 % success

Add

Remove

Edit

Activar Windows
Ve a Configuración para activar Windows.

Editar mazo seleccionado

< QuizCard

fresa

My decks collection

Open

holiii

Spanish - Hindi

6 cards

40 % success

Update

Cancel

Activar Windows
Ve a Configuración para activar Windows.

Abrir mazo seleccionado

< QuizCard

fresa

holiii

Spanish - Hindi

Invite a friend!

New question3333 - New answer3333

42 % success

New question1111 - New answer1111

20 % success

New question4444 - New answer4444

31 % success

New question0000 - New answer0000

45 % success

Add

Remove

Edit

Activar Windows

Ve a Configuración para activar Windows.

Seleccionar carta del mazo

< QuizCard

fresa

holiii

Spanish - Hindi

Invite a friend!

New question3333 - New answer3333

42 % success

New question1111 - New answer1111

20 % success

New question4444 - New answer4444

31 % success

New question0000 - New answer0000

45 % success

Add

Remove

Edit

Activar Windows

Ve a Configuración para activar Windows.

Editar carta seleccionada

< QuizCard

fresa

holiii

Spanish - Hindi

Invite a friend!

New question3333 - New answer3333

42 % success

New question1111 - New answer1111

20 % success

New question4444 - New answer4444

31 % success

New question0000 - New answer0000

45 % success

Update

Cancel

Activar Windows

Ve a Configuración para activar Windows.

Añadir un usuario a un mazo

< QuizCard

fresa ▾

holiii

Email

YourFriend@email.com

Invite

Cancel

New question3333 - New answer3333

42 % success

New question1111 - New answer1111

20 % success

New question4444 - New answer4444

31 % success

New question0000 - New answer0000

45 % success

Add

Remove

Edit

Activar Windows

Ve a Configuración para activar Windows.

Sección 'Repository'

Buscar por idiomas

< QuizCard

fresa ▾

Repository

Native language:

Spanish ▾

Target language:

Hindi ▾

Search

Resultado de la búsqueda

< QuizCard

fresa ▾

Repository

holiii

Spanish - Hindi

6

Contenido de un mazo elegido

< QuizCard

fresa ▾

Repository

(holiii)

New question3333 ~ New answer3333

New question1111 ~ New answer1111

New question4444 ~ New answer4444

New question0000 ~ New answer0000

New question5555 ~ New answer5555

Copy

Cancel

Activar Windows
Ve a Configuración para activar Windows.

Seleccionar mazo propio al que importa cartas

< QuizCard

fresa ▾

Repository

(Choose deck to copy in)

holiii

Spanish ~ Hindi

6

Cancel

Activar Windows
Ve a Configuración para activar Windows.

Sección ‘My Statistics’

Mazo de ejemplo con 50% o más aciertos (sin/con Hover)

< QuizCard

fresa ▾

My statistics

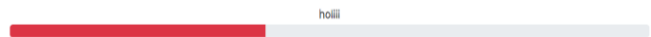
holiii

My statistics

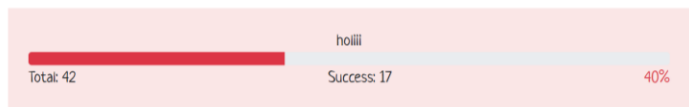


Mazo de ejemplo con menos del 50% de aciertos (sin/con hover)

My statistics



My statistics



NavBar

MY DECKS

Build your own decks collection, and customize your study!

Capítulo 5 . IMPLEMENTACIÓN DEL SISTEMA

5.1. ESTÁNDARES Y NORMAS SEGUIDOS

- Uso de TypeScript como lenguaje tipado (a diferencia de JavaScript)
- Inyección de dependencias para la implementación del servicio Firestore (permite implementar el servicio de operación con cualquier BD sin necesidad de modificar la lógica del front-end)
-

5.2. LENGUAJES DE PROGRAMACIÓN

- TypeScript
- CSS
- HTML

5.3. HERRAMIENTAS Y PROGRAMAS USADOS PARA EL DESARROLLO

- VisualStudio Code
- Google Cloud Console
- Google Chrome

5.4. CREACIÓN DEL SISTEMA

5.4.1. Problemas Encontrados

La optimización del manejo de observables fue compleja, debido a que inicialmente se pretendía cargar toda la información de los mazos/cartas del usuario desde la BD al inicio del juego para permitir el juego incluso en casos de pérdida de señal de internet.

Esto requería realizar consultas asíncronas anidadas a la BD: dentro de la consulta para obtener los mazos debían realizarse dos consultas: para obtener la información de la colección de estadísticas y para obtener la información de la colección de las cartas. Esta última consulta a su vez debía realizar nuevas consultas anidadas: para obtener la colección de estadísticas de cada carta

Finalmente, se optó por cargar los datos en tiempo real, permitiendo emplear los observables de manera independiente en el momento concreto en que la aplicación requiera emplearlos o visualizarlos. Con esto se consigue además optimizar el volumen de datos descargados desde la BD.

5.4.2. Descripción Detallada de las Clases

Las clases se han modelado de la manera más simple posible para dejar el grueso de la funcionalidad de la aplicación a la lógica de juego.

La BD es una colección de mazos. La clase mazo (**Deck**) está compuesto además de por sus atributos sencillos (ID, nombre, idioma de estudio e idioma nativo) por dos subcolecciones: una colección con las estadísticas de cada uno de los usuarios que juega con dicho mazo y una colección con las cartas que lo componen.

La clase carta (**Card**) está compuesta por la pregunta y respuesta y por su ID, y por una subcolección con las estadísticas de cada uno de los usuarios que juega con el mazo al que pertenece dicha carta.

La clase estadística (**Statistic**) está compuesta por el ID del usuario al que corresponde, el número de veces totales respondido y el número de veces acertado. Esta clase no cuenta con un ID propio pues no se necesitan identificar de manera concreta en ningún momento en el juego. Sólo se accede a sus instancias a través del mazo o carta al que pertenecen, y nunca se accede a la información que contienen de forma independiente si no es por medio de una consulta a ese mazo o carta.

Capítulo 6 . MANUALES DEL SISTEMA

6.1. MANUAL DE INSTALACIÓN

Al tratarse de una aplicación web no requiere de ninguna aplicación especial. Sencillamente acceder a la URL correspondiente.

6.2. MANUAL DE USUARIO

La aplicación no requiere de ninguna explicación previa para ser utilizada. Las diferentes acciones que puede realizar el usuario aparecen detalladas en el apartado 'Casos de uso' del proyecto.

Capítulo 7 . CONCLUSIONES Y AMPLIACIONES (OPTATIVO)

7.1. CONCLUSIONES

Tanto JavaScript como TypeScript suponen dos lenguajes de programación muy ágiles y flexibles para el desarrollo web, por permitir el funcionamiento de las aplicaciones aún cuando ciertas partes de la misma pudieran llegar a ocasionar fallo ante determinadas acciones del usuario. De este modo, se evita que la aplicación quede totalmente inutilizada por errores de programación que podrían no llegar a ocasionarse nunca si tenemos en cuenta que el usuario podría no hacer uso de las partes problemáticas de la app.

A diferencia de lo que cabría esperar para un software empresarial o de gestión para el cual un lenguaje con unos estándares más estrictos como Java o .NET sería más indicado, en el caso de las aplicaciones web los lenguajes más permisivos frente a los errores de programación permiten ofrecer una mejor experiencia de usuario al mantener los componentes no problemáticos funcionales a pesar de posibles problemas en el conjunto de la aplicación.

7.2. AMPLIACIONES

La principal ampliación contemplada sería la adaptación para dispositivos móviles con IONIC.

Como parte necesaria para completar al 100% la aplicación estaría el back-end de registro de usuarios, y su correspondiente BD. La aplicación se ha desarrollado con vistas a que se pudieran desarrollar en cualquier tecnología deseada.

Capítulo 8 . REFERENCIAS BIBLIOGRÁFICAS

Aspectos generales:

<https://docs.npmjs.com/uninstalling-packages-and-dependencies>

https://www.w3schools.com/jsref/dom_obj_event.asp

<https://appmakers.dev/bcp-47-language-codes-list/>

https://developer.mozilla.org/en-US/docs/Web/API/SpeechRecognition#browser_compatibility

Integración Firebase Firestore con Angular:

<https://medium.com/factory-mind/angular-cloud-firestore-step-by-step-bootstrap-tutorial-ecb96db8d071>

<https://medium.com/factory-mind/angular-firebase-typescript-step-by-step-tutorial-2ef887fc7d71>

<https://medium.com/factory-mind/cloud-firestore-angular-cud-operations-and-transactions-deep-dive-bcbafd8a7536>

<https://angular-templates.io/tutorials/about/angular-crud-with-firebase>

<https://www.techiediaries.com/angular-firebase/angular-9-8-firestore-database-crud-tutorial/>

<https://github.com/angular/angularfire>

Documentación Firebase Firestore:

<https://fireship.io/snippets/firestore-increment-tips/>

<https://firebase.google.com/docs/firestore/data-model>

<https://firebase.google.com/docs/firestore/query-data/get-data>

<https://firebase.google.com/docs/firestore/query-data/listen>

<https://firebase.google.com/docs/reference/js/firebase.firestore.CollectionReference>

<https://firebase.google.com/docs/firestore/manage-data/delete-data>

<https://firebase.google.com/docs/firestore/manage-data/enable-offline>

<https://firebase.google.com/docs/firestore/query-data/indexing>

<https://firebase.google.com/docs/firestore/query-data/listen#events-metadata-changes>

<https://www.youtube.com/playlist?list=PLI-K7zZEsYLLuG5MCVEzXAQ7ACZBCuZgZ>

Capítulo 9 . APÉNDICES

9.1. GLOSARIO Y DICCIONARIO DE DATOS

Web App: Programa desarrollado en un lenguaje interpretable por los navegadores web en la que se la ejecución la lleva a cabo el propio navegador. Sus principales ventajas son el uso de un cliente ligero (navegador web), la independencia del sistema operativo, y la facilidad del mantenimiento y actualización de la aplicaciones sin necesidad de distribuir e instalar *software* a un gran número de usuarios.

SPA: Siglas en inglés de Single Page Application o Aplicación de Página Única. Aplicación web constituida por una sola página. Los códigos de HTML, JavaScript, y CSS se cargan una sola vez y los recursos necesarios se cargan dinámicamente como respuesta a las acciones del usuario. De este modo, la página no tiene que cargarse de nuevo en ningún momento consiguiendo una experiencia de usuario más fluida.

Lenguaje tipado: Lenguaje de programación en que asignado el valor de una variable de un tipo concreto, no se puede usar como si fuera de otro tipo distinto a menos que se haga una conversión

IDE: Siglas en inglés de Integrated Development Environment o Entorno de Desarrollo Integrado. Aplicación informática que proporciona servicios integrales para facilitar el desarrollo de software. Normalmente consiste en un editor de código fuente, herramientas de construcción automáticas y un depurador (debugger). Muchos tienen auto-completado inteligente de código (IntelliSense), un compilador, o un intérprete.

Inyección de dependencias: Patrón de diseño orientado a objetos en el que se suministran objetos a una clase en lugar de ser la propia clase la que cree dichos objetos. Esos objetos cumplen funciones necesarias dentro de nuestras clases para poder funcionar (dependencia). Estos objetos serán suministrados por otra clase 'contenedora' que inyectará la implementación concreta deseada de dichos objetos.

API: Siglas en inglés de Application Programming Interface. Conjunto de definiciones y protocolos utilizado para desarrollar e integrar el software de las aplicaciones, permiten intercomunicar servicios abstrayéndose de la forma en que dichos protocolos estén implementados.

Capítulo 10 ÍNDICE

1. MEMORIA DEL PROYECTO	3
Resumen de la motivación	3
Objetivos y alcance del proyecto	3
2. INTRODUCCIÓN	5
Justificación del proyecto	5
Estudio de la situación actual	6
Evaluación de las alternativas	6
Aspectos teóricos	7
3. ANÁLISIS	9
Requisitos	9
Requisitos no funcionales	9
Requisitos funcionales	10
Especificación casos de uso	10
4. DISEÑO DEL SISTEMA	29
Arquitectura del sistema	29
Diagrama de paquetes	29
Diseño de clases	29
Diagrama de paquetes	31
Diseño de bases de datos	32
Diseño de la interfaz	34
5. IMPLEMENTACIÓN DEL SISTEMA	45
Estándares y normas Seguidos	45
Lenguajes de programación	45
Herramientas y programas usados para el desarrollo	45
Creación del sistema	45
Problemas encontrados	45
Descripción detallada de las clases	46
6. MANUAL DEL SISTEMA	47
Manual de instalación	47
Manual de usuario	47
7. CONCLUSIONES Y AMPLIACIONES	49
Conclusiones	49
Ampliaciones	49

8. REFERENCIAS BIBLIOGRÁFICAS	51
9. APÉNDICES	53
Glosario y diccionario de datos	53
10. ÍNDICE	55