



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



TFG del Grado en Ingeniería
Informática

Seguimiento de
requerimientos para gestionar,
medir, evaluar y mejorar
documentación técnica



Presentado por Pablo Ahíta del Barrio
en Universidad de Burgos — 9 de julio de 2024
Tutor: M^a Belén Vaquerizo García



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D^a. M^a Belén Vaquerizo García, profesora del departamento de Ciencia de la Computación e Inteligencia Artificial.

Expone:

Que el alumno D. Pablo Ahíta del Barrio, con DNI 71566290L, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado Seguimiento de requerimientos para gestionar, medir, evaluar y mejorar documentación técnica.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 9 de julio de 2024

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. nombre tutor

D. nombre co-tutor

Resumen

En este proyecto de final de grado se ha desarrollado una aplicación Android con el objetivo de sustituir a un antiguo script de nombre *OTEA*, que almacena localmente en cada ordenador la información sobre los test de indicadores o características de calidad de vida realizados por la *Fundación Miradas*.

La forma en la que se trasladaba el programa entre diferentes computadores era mediante un CD-ROM que tenía que llevar a mano siempre la persona encargada de hacer el test de indicadores, tecnología que se ha sustituido por una arquitectura cliente-servidor, por lo que la funcionalidad de esta aplicación Access se traslade a una aplicación para Android que se conecta a una aplicación web desplegada en Azure que a su vez se comunica con la base de datos para realizar todas las consultas que se le solicitan.

En cuanto a la arquitectura, el cliente es la propia aplicación de Android desarrollada en *Java*, con su interfaz gráfica y su código interno para manejar las peticiones HTTP al servidor mediante *Retrofit* y *OkHttp* con la ayuda de *java.util.concurrent* que ejecuta esas órdenes en un hilo diferente al de la propia aplicación y para realizar toda la funcionalidad esperada dependiendo del tipo de usuario que maneje dicha aplicación. Mientras tanto el servidor se ha desarrollado en *C#*, utilizando para el manejo de las peticiones HTTP la API de *ASP.NET*, la cual se encarga de recibir las peticiones por parte del cliente y de enviar la respuesta del endpoint solicitado, utilizando *Entity Framework* para comunicarse con la base de datos y el paquete *Newtonsoft.Json* para transformar a JSON el objeto y serializar sus parámetros para que el cliente mediante *Retrofit* los deserialice y los utilice para crear las instancias correspondientes.

Los tipos de usuarios distinguidos por la aplicación ayudan a distinguir correctamente las acciones que cada uno de ellos puede realizar en la aplicación. En primer lugar tenemos los **administradores** que realizan las pruebas de indicadores, los **directores de organización evaluada** que gestionan su organización y los **usuarios de organización**.

Descriptores

Android, Azure, Azure SQL, C#, cliente-servidor, evidencias, indicadores, JSON, SQL-Server, servicio web.

Abstract

In this final degree project an Android app has been developed with the goal of substituting an old Access Script named *OTEA*, that stores locally in every computer all the information about indicators or life quality features realized by *Fundación Miradas*.

The way used to move the softwares between computers was using a CD-ROM that the person that realizes the indicator test had to take that with him, technology substituted by an architecture client-server, so the old Access application functionality was finally to an Android app that communicates with a web-app deployed in Azure that communicates with a Azure SQL database to realize all the queries ordered by the client.

Regarding the architecture, the client is the proper Android app developed in *Java*, including its graphic interface and the internal code that controls the HTTP requests using *Retrofit* and *OkHttp*, both being helped by *ASyncTask* that runs all the calls to the server in an app-separated thread, becoming in a huge help to realize all the functionality depending of the user's type. Meanwhile the server was developed using *C#*, using *ASP.NET* to control all the HTTP requests that are received from the client and then sending the requested endpoint's response, using for that the *Entity Framework* to communicate with the database and the *Newtonsoft.Json* to convert to Json the object and serialice its parameters before the client deserialices it using *Retrofit* and then uses it to build the corresponding instances.

The types of users distinguished by the application help to distinguish correctly the actions that each one of them can carry out in the application. First of all we have the **administrators** who are in charge of carry out different evaluations of indicators, the **evaluated organization directors** that manage their own organization and the *organizations users*.

Keywords

Android, Azure, Azure SQL, C#, client-server, evidences, indicators, JSON, SQL-Server, web-service.

Agradecimientos

En primer lugar agradezco a mi familia, en especial a mis padres **Diego y Yolanda** y a mis **abuelos Lola y Salva**, por ser el principal motor de mi vida en todo momento gracias a su constante apoyo y cariño, en el que también están mis mejores amigos de toda la vida **Antonio y Adrián**, con quienes he aprendido el significado de la verdadera amistad.

Asimismo quiero expresar mi gratitud a los docentes que me han apoyado a lo largo de mi etapa universitaria, a mi tutor del trabajo de final de grado del curso pasado **Pedro Luis Sánchez** por su constante guía y apoyo en este proyecto y a mi tutora de este trabajo de final de máster **María Belén Vaquerizo García** por darme la oportunidad de poder continuar con este proyecto como trabajo de final de máster para la conclusión de dicho proyecto, aportándome su visión sobre el factor diferencial de este proyecto, el sistema experto. También agradezco a la *Fundación Miradas* su plena confianza en mí y su involucración en el proyecto, a su director **Miguel Gómez**, a su analista **Fernando Terradillos** y **Jose Luis Cuesta**, director de la *Cátedra Miradas por el Autismo*.

A su vez quiero agradecer a **Sonia Rodríguez** su apoyo en todo momento y por acompañarme y guiarme a lo largo de mi vida académica y personal.

Asimismo muestro mi agradecimiento a los docentes que me han apoyado y me han animado a lo largo de todas las etapas educativas que he ido superando hasta la fecha.

También muestro mi agradecimiento a **Natividad de Juan** su constante orientación, respaldo y asesoramiento durante toda mi etapa universitaria.

A su vez muestro mi agradecimiento a **Cristina Barriuso** por su fe en mí y a **Jennifer Terceño** por su constante apoyo durante todos estos años.

Por último, quiero recordar y agradecer al profesor **Joaquín Seco** por haberme aportado sus conocimientos y experiencia para este proyecto.

Índice general

Índice general	iv
Índice de figuras	vi
Índice de tablas	vii
Introducción	1
Objetivos del proyecto	3
2.1. Objetivos generales	3
2.2. Objetivos específicos	4
Conceptos teóricos	7
3.1. Guía de indicadores extensa	7
3.2. Guía de indicadores reducida	52
Técnicas y herramientas	65
4.1. <i>Microsoft Azure</i>	65
4.2. Entornos de desarrollo utilizados	66
4.3. Lenguajes de programación y herramientas utilizadas	72
4.4. Sistema experto	81
Aspectos relevantes del desarrollo del proyecto	93
5.1. Ciclo de vida utilizado	93
5.2. Fases de análisis, diseño e implementación	95
5.3. Pruebas	114
Trabajos relacionados	119

<i>ÍNDICE GENERAL</i>	v
Conclusiones y Líneas de trabajo futuras	123
Bibliografía	125

Índice de figuras

3.1. Tabulación de datos en evaluación completa	19
3.2. Tabla de puntuaciones por puntuación y por colores	20
3.3. Tabla de rangos de puntuación total	20
3.4. Rangos de puntuación en evaluación completa	22
3.5. Rangos de puntuación en evaluación simple	53
3.6. Tabulación simple de datos	63
4.1. Estructura de un sistema experto	83
5.1. Uso de <i>Trello</i> para el proyecto	94
5.2. Estructura básica de la arquitectura <i>Modelo-Vista-Controlador</i> .	98
5.3. Resultados de prueba de indicadores de prueba	118

Índice de tablas

Introducción

- El repositorio de *GitHub* se encuentra en el siguiente enlace:[10]
- El vídeo de presentación se encuentra en el siguiente enlace:[12]. También se encuentra en la raíz del repositorio de *GitHub*.
- El vídeo de demostración se encuentra en el siguiente enlace:[11]. También se encuentra en la raíz del repositorio de *GitHub*.

En este proyecto final de máster se ha desarrollado una aplicación compatible con Android denominada *OTEA (Organizaciones que prestan apoyo a personas con Trastorno del Espectro Autista)* para la fundación especializada en trastorno del espectro autista con sede en la ciudad de Burgos denominada *Fundación Miradas*.

El proyecto como tal consiste en el paso de una aplicación en un fichero de Microsoft Access grabado en un CD-ROM, donde se guardaban todos los datos y registros, y se pasaba de mano en mano a cada uno de los ordenadores que lo necesitan. El cambio a una implementación más moderna de esta aplicación se antoja necesario, ya que en primer lugar el uso de los discos ópticos ya no es habitual en la informática de la actualidad, puesto que cada vez más equipos carecen de una bandeja compatible con estos discos, a parte de que la versión utilizada data del año 2009 sobre una versión del paquete de *Microsoft Office 2007*, quedando dichas tecnologías como totalmente obsoletas.

Una solución a este problema nos la proporciona la computación en la nube, el uso masificado de los dispositivos móviles que permite la existencia de las denominadas aplicaciones multidispositivo y el auge de las implementaciones cliente-servidor, aspectos que quedan perfectamente cubiertos gracias a la implementación de una aplicación web y de una base de datos en *Microsoft*

Azure, permitiendo la comunicación entre las diferentes tablas de la base de datos y la web app que recibe las peticiones correspondientes a cada endpoint que realiza para posteriormente devolver la consulta de la base de datos correspondiente y enviarla a posteriori al cliente. En cuanto a la aplicación, se ha decidido que se va a implementar en primer lugar para Android, debido a que es la plataforma de dispositivos móviles más utilizada en el mercado a nivel mundial por el bajo costo de los dispositivos que cuentan con el mencionado sistema operativo en comparación con su competencia, cubriendo una cuota de mercado importante.

Objetivos del proyecto

2.1. Objetivos generales

En cuanto a los objetivos a cumplir en este proyecto, se tienen que tener en cuenta los cuatro principios que debe tener todo software, siendo éstos el *control*, la *comodidad*, la *eficiencia* y la *evolución*, sirviendo de base para poder mencionar los objetivos que se cumplen y se tienen que seguir cumpliendo en la aplicación de *OTEA*, para garantizar el mejor funcionamiento posible de la misma dentro de cada uno de esos cuatro principios. Por lo tanto, los objetivos a esperar dentro del software son los siguientes:

- **Objetivos relacionados con el principio del control:** La aplicación *OTEA* debe garantizar que los usuarios tengan un control total sobre las acciones que realizan dentro de la aplicación y los resultados que tengan de las mismas. En este caso se busca que los usuarios de la Fundación Miradas puedan realizar los test de indicadores marcando las evidencias que se cumplen, para luego hacer que los usuarios de las organizaciones evaluadas puedan observar los resultados de la puntuación total de cada test y sus respectivos gráficos. Además los usuarios deben tener el poder de personalizar la propia aplicación en la medida de lo posible, como elegir el idioma de la misma, algo que se consigue gracias a que se ha realizado la internacionalización a tres idiomas (español, inglés y francés).
- **Objetivos relacionados con el principio de la comodidad:** La aplicación *OTEA* debe ser una aplicación intuitiva y fácil de aprender a manejar, pudiendo ser utilizado por usuarios de todos los niveles, desde los usuarios casuales hasta los usuarios expertos. En caso de que

se necesite ayuda, se tiene que ofrecer un manual de instrucciones de la misma, priorizando la existencia de vídeos junto con un manual escrito. La aplicación también tiene que estar preparada para que tenga el soporte para los tres tipos de usuarios: **administradores**, **usuarios de organizaciones evaluadas** y **usuarios de la *Fundación Miradas***, teniendo cada uno de los usuarios sus funcionalidades muy bien marcadas desde el principio.

- **Objetivos relacionados con la eficiencia:** La aplicación *OTEA* debe garantizar unos tiempos de respuesta en las peticiones a la base de datos en formato HTTP lo más rápidas y eficientes posible, haciendo que no se consuman una cantidad enorme de recursos en el dispositivo, en cuanto a espacio del disco duro y uso del procesador y de la memoria RAM. Gracias al correcto manejo de la aplicación en aspectos de hardware, se espera que sea de comportamiento fluido y con los tiempos de espera a las respuestas de las peticiones HTTP lo más cortos posible.
- **Objetivos relacionados con la evolución:** La aplicación *OTEA* debe ser una aplicación que sea siempre susceptible a diferentes cambios y mejoras en la funcionalidad de la misma, adaptándose siempre al avance constante de la computación en la nube y de las tecnologías móviles. Al desarrollador no le tiene que temblar el pulso para poder tomar decisiones arriesgadas que puedan desembocar en la adición de dichas mejoras y características adicionales, cumpliendo así con el carácter ambicioso que tiene la *Fundación Miradas* para realizar su cometido.

2.2. Objetivos específicos

El objetivo específico principal de esta ampliación del proyecto es presentar un producto que ya pueda ser utilizado a nivel nacional y a nivel internacional, centrándose dichas pruebas en el mercado hispanohablante que utilizará la versión final de la aplicación cuando esta sea lanzada.

De forma más desarrollada, se pretende conseguir lo siguiente en esta segunda parte del proyecto:

- En primer lugar, se espera que la aplicación sea una herramienta de utilidad para los diferentes profesionales que tratan a personas dentro del espectro autista, con la ayuda de la Fundación Miradas y de los indicadores ubicables en el libro del Dr. José Luis Cuesta Gómez.

- Más adelante, se deben corregir los errores mencionados en el documento del trabajo de final de grado del curso pasado, los cuales impedían que la aplicación funcionase de una manera óptima.
- Posteriormente, se espera que en esta fase del proyecto se utilice la inteligencia artificial como herramienta clave para el trabajo de *Fundación Miradas*. Como propuesta de la tutora de este proyecto y coordinadora de este máster, la Dra. M^a Belén Vaquerizo García, se implementará un **sistema experto** dentro de la dinámica cliente-servidor de este proyecto, implementándose este en el lado del servidor mediante el motor de inferencia de *NRules*.
- Adicionalmente se espera ampliar la seguridad de todo el entorno de la aplicación cliente-servidor, mediante el uso de un token bearer con información sobre el email del usuario y sobre el tipo de usuario que es, restringiendo así los endpoints con información sensible a los usuarios registrados.
- Se espera de igual manera mejorar la persistencia y la eficiencia de los endpoint clave, como el de inicio de sesión y aquellos endpoints que devuelvan listas, utilizando variedad de técnicas en el lado del cliente para su correcto procesamiento.
- Se espera también una visualización completa de los resultados, tanto en informes como en la propia aplicación, permitiendo así una exploración intuitiva y amigable.
- Se espera contar con una herramienta de administración, en formato de aplicación web, para el manejo de los registros de los diferentes usuarios, salvo en el caso de los directores de organizaciones externas, cuya aceptación de la solicitud es automática en el registro de una organización, ya que se provee el email del mismo director en dicho proceso.
- Se espera mejorar visualmente la aplicación con un uso más eficiente y visualmente atractivo de las diferentes herramientas de Android Studio.
- Se espera que los indicadores, en su versión definitiva, ya estén disponibles en la base de datos, al igual que utilizar la nube de *Azure* de Fundación Miradas.

Conceptos teóricos

En dicho apartado se va a desarrollar en que consiste el proyecto anteriormente mencionado denominado *OTEA*, el cual se ha obtenido de la guía escrita por el profesor de la Universidad de Burgos *Jose Luis Cuesta Gómez*.^[13] Dicho capítulo sirve como punto de referencia para poder mostrar el trabajo que se ha ido añadiendo poco a poco en la aplicación, y la funcionalidad que se espera de la misma. Se han desarrollado dos guías de indicadores, la **guía de indicadores extensa** y la **guía de indicadores reducida.**, cuyo procedimiento está basado en el Método Delphi.

3.1. Guía de indicadores extensa

Introducción

El modelo de calidad de vida es un referente para la planificación y el desarrollo de servicios de apoyo en ámbitos como la educación, la salud o los servicios sociales. Este modelo ha favorecido el desarrollo de propuestas de evaluación y la inclusión de planes de mejora en numerosos sectores institucionales, programas y organizaciones.

En el ámbito de la discapacidad, los modelos más actuales referidos a la calidad abarcan la evaluación del impacto de los servicios en la persona. Además de los aspectos más formales sobre gestión y organización de servicios, se incorpora la valoración de los resultados personales desde una visión multidimensional, que atiende a todas las áreas, ámbitos y contextos de la vida de la persona, y que integra dos perspectivas: objetiva (cuestiones observables y fácilmente medibles) y subjetiva (el grado de satisfacción percibido por cada persona), que abarca el conocimiento del ajuste entre las condiciones de vida y las aspiraciones y expectativas personales, aspectos

que van unidos a un enfoque de intervención basado en la ética y en el modelo de derechos.

Entre los trabajos de investigación sobre calidad de vida, destacan los realizados por *Schalock (2000)*, *Verdugo, (2006)*, *Schalock y Verdugo (2006)*, *Verdugo y Schalock (2007)*, *Schalock et al. (2021a)*, *Schalock et al. (2021b)*, *Verdugo et al. (2021)* y *Schalock y Verdugo (2021)*, que han consolidado un marco teórico ampliamente consensuado en torno a este concepto. Calidad de vida se asocia a percepción individual, a sentimientos de bienestar, de inclusión, de oportunidades de desarrollo personal... , pero también es un concepto que va unido a condiciones y contextos de vida.

Debido a las dificultades que implica el análisis del bienestar subjetivo en las personas con Trastorno del Espectro Autista (TEA), se plantea el diseño de una guía centrada en la dimensión objetiva, con un enfoque ecológico, contemplando el contexto desde el cual se planifican y coordinan los planes de desarrollo de la persona y definiendo qué condiciones deben reunir éstos para promover calidad de vida.

Este instrumento se configura como una guía de referencia para la planificación y la evaluación de los programas y servicios de apoyo para personas con autismo, cuyos resultados pueden orientar también, si fuera necesario, para la elaboración de Planes de Mejora.

Como premisa, debemos abordar la calidad de vida de las personas con autismo desde tres perspectivas:

- **Calidad en la intervención:** Una intervención basada en evidencias implica, en la línea de lo que propone *Research Autism (2018)*, conocer si está avalada por la investigación, siempre que responda a parámetros científicos, y si esta permite asegurar que es eficaz y puede ser recomendada.

Por otra parte, existen criterios basados en posicionamientos internacionales, y en la experiencia y el juicio de expertos que, sin referirse a ningún tratamiento o intervención de forma específica, nos permiten discriminar una buena práctica. (*Guldborg, 2017, AETAPI, 2011, Charman et al., 2011, Tamarit, 2010, Güemes et al. 2009. Fuentes et al., 2006*):

- Implicación de la persona con autismo y, en su caso, de las personas más significativas de su entorno, asegurando su participación en el diseño, desarrollo y evaluación de los programas de apoyo.
- Evaluación previa de destrezas y puntos débiles en las diferentes áreas de desarrollo y de funcionamiento adaptativo.
- Valoración de las dificultades en la autorregulación conductual, basando el análisis funcional y la intervención en los principios y prácticas del apoyo conductual positivo.
- Personalización de los contenidos del programa, así como de los apoyos destinados a llevarlo a cabo.
- Empleo de estrategias eficaces y sistemáticas de enseñanza, definiendo metas específicas y planes para lograrlas, y favoreciendo los aprendizajes en contextos naturales e inclusivos.
- Generalización de aprendizajes a través de la enseñanza de habilidades con validez ecológica, en entornos naturales y rutinas diarias.
- Desarrollo del plan de intervención, estableciendo claramente las metas y objetivos. Concretamente:
 - Priorizando metas que conjuguen criterios personales y de contexto, asegurando su funcionalidad y adecuación a la edad cronológica (por ejemplo, comunicar emociones, pedir ayuda cuando sea necesario, hacer elecciones, iniciar comentarios espontáneos a otras personas implicadas en la actividad, establecer relaciones significativas).
 - Favoreciendo especialmente las habilidades de comunicación espontáneas y funcionales, así como la competencia social, proporcionando oportunidades y herramientas (por ejemplo, sistemas alternativos de comunicación) para aplicarlas funcionalmente en el contexto natural, de manera cotidiana.
 - Fomentando la participación activa en actividades inclusivas favoreciendo las oportunidades para disfrutarlas
 - Instaurando procesos de evaluación, innovación y mejora continua que permitan el enriquecimiento continuado de las actuaciones, y la adaptación de las mismas a las necesidades cambiantes de las personas con autismo a lo largo de su ciclo vital.

Igualmente, como afirma AETAPI (2014), resulta imprescindible incorporar en la práctica profesional un enfoque basado en derechos,

más teniendo en cuenta que en la intervención y relación cotidiana con las personas con autismo, podemos encontrarnos a diario situaciones en las que los profesionales no somos conscientes de que con nuestro comportamiento o actitud personal contribuimos a que la persona no disfrute de las mismas oportunidades que los demás.

- **Calidad en los servicios de apoyo:** Garantizar que las organizaciones y servicios de apoyo aseguren condiciones directamente relacionadas con la calidad de vida de las personas con TEA. Para ello debemos contar con herramientas de evaluación y planificación, que nos permitan evaluar en qué medida estos promueven calidad de vida, tal como afirman *Cuesta (2009)* y *AETAPI (2011)*, describiendo los principales criterios a tener en cuenta para que una organización promueva calidad de vida en las personas con autismo:

- ***En relación a organización de los servicios:***

- Existe una red de servicios: diagnóstico y evaluación, atención temprana, educación, orientación, formación en la etapa adulta, formación e inclusión laboral, vivienda, ocio, tiempo libre y deportes. . . Los servicios cubren las distintas etapas vitales y los diferentes ámbitos de la vida de la persona con autismo.
- Se presta apoyo al entorno de referencia, significativo en la vida de la persona con autismo.

- ***En relación al enfoque integral de la intervención:***

- Los programas dan respuesta, de forma personalizada, a todas las necesidades de la persona y promueven su desarrollo en todas las áreas.
- Se favorece la actividad laboral y el empleo de las personas con autismo.
- Se promueve la inclusión educativa y social.
- La persona con autismo tiene un papel activo, colabora en el diseño, desarrollo y evaluación de los programas de intervención y apoyo.
- Existe una coherencia entre el planteamiento teórico y la práctica.
- Los entornos están adaptados y son accesibles para la persona con autismo (Control de estimulación ambiental, estructuración espacio-temporal, condiciones de ratio y agrupamientos adaptados a las necesidades individuales. . .), asegurando que

sean comprensibles para todas las personas, garantizando condiciones de seguridad y la participación más autónoma posible.

- La estructura de los servicios de apoyo es funcional, flexible y adaptada al perfil de cada persona con autismo.
 - La organización está centrada en las personas y permite adaptarse a las necesidades cambiantes.
 - La organización cuenta con un equipo multidisciplinar, especializado en la atención a personas con autismo, con un enfoque orientado al desarrollo personal continuo, que trabaja de forma coordinada e interdisciplinar.
 - Los programas se basan en protocolos de buena práctica en el ámbito del autismo y tienen como referencia parámetros de eficacia y eficiencia contrastada a través de la ciencia y la experiencia.
 - Los programas son personalizados y favorecen aspectos clave como la inclusión en la comunidad, el bienestar físico y emocional, etc.
 - Se cuenta con protocolos específicos de carácter preventivo, que garanticen la seguridad ante situaciones de riesgo, abuso y/o violencia.
- ***En relación a la creación, transferencia y difusión del conocimiento:***
 - La organización cuenta con un sistema de gestión del conocimiento, interno y externo.
 - Se promueve la actualización científica y formación continua de los profesionales en aspectos clave relacionados con el autismo.
 - La organización promueve la investigación colaborando con centros o equipos de investigación.
 - Se promueven y difunden buenas prácticas.
 - Se desarrollan acciones de formación interna y externas con profesionales de distintas áreas disciplinares que impacten de manera directa e indirecta a las personas con autismo
 - Se dispone de un programa de formación, apoyo y seguimiento a las familias o personas significativas en la vida de la persona con autismo.
 - ***En relación al trabajo en red:***

- Se colabora con otras entidades del sector.
- La Organización participa activamente en redes regionales, nacionales, internacionales de autismo.
- Se trabaja de forma activa en colaboración con sectores clave: servicios sociales, sanidad, educación. . .
- Se promueven y desarrollan proyectos de carácter internacional.

- ***En relación a los procesos de mejora continua:***

- La organización tiene implantado un sistema interno y externo de evaluación de sus procesos y resultados, tanto desde el punto de vista de la gestión como de los resultados personales.
- La organización aplica regularmente sistemas de evaluación que miden su impacto en la calidad de vida de las personas con autismo.
- La intervención se basa en un código ético que contribuye a impulsar, promover y desarrollar normas, procedimientos y buenas prácticas implicando a todos los grupos de interés.
- Los servicios promueven la calidad de vida, atendiendo a cada una de sus dimensiones.
- Se cuidan las relaciones laborales, existen planes de igualdad, de gestión de la diversidad, conciliación, etc. En la organización se contempla, de manera explícita, la perspectiva de los derechos, actuando bajo el marco de la igualdad y la diversidad de las personas con autismo, con el fin de no establecer discriminaciones o desigualdad en las oportunidades, en el trato y en el acceso al servicio, por razón de género, edad, raza, creencias, situación de salud, o cualquier otra circunstancia personal.
- Se establecen canales para la propuesta de mejoras por parte de los diferentes grupos de interés de la organización.
- Se desarrollan procesos de mejora continua con canales de comunicación y recepción de propuestas de los diversos sectores, áreas o grupos que componen a la organización
- Las actuaciones responden a un Plan Estratégico previamente diseñado y consensuado entre los diferentes miembros de la organización.

- ***En relación al impacto social:***

- Se diseñan e implementan estrategias de comunicación sobre acciones de la organización con impacto social.
- Se impulsan y desarrollan acciones de sensibilización, concienciación e involucramiento que tengan impacto social.
- Se promueve una imagen positiva que promueva la eliminación de los mitos, calificaciones negativas y estigmatizantes en relación a las personas con autismo
- Se planifica, mide y evalúa el impacto social de la organización.
- ***En relación al compromiso social:***
 - Se colabora y presta apoyo a otras entidades y profesionales.
 - Se colabora con países y entidades en desarrollo en el ámbito del autismo.
 - Se realiza una acción positiva hacia otras organizaciones sociales, empresas locales y empresas responsables a la hora de adquirir productos o servicios.
 - Se cuenta con una política de sostenibilidad y se establece un plan específico para su implementación y gestión abarcando todas las áreas de la organización y servicios.
 - La organización tiene una política de transparencia.
 - Se contempla el desarrollo de las personas voluntarias como elemento estratégico de gestión.
- **Calidad de vida personal:** Es imprescindible utilizar herramientas y metodologías que faciliten la obtención de información, directa o indirecta, de la persona con autismo, para conocer qué cosas considera importantes y cuál es su nivel de satisfacción acerca de sus condiciones de su vida. Esta información es clave para poder orientar el diseño de los planes de desarrollo personal y servir de referencia para establecer prioridades en la organización (*Vidriales et al., 2017, Vidriales et al., 2017*). Al mismo tiempo, se sugiere contrastar esta información con la aportada por familias y profesionales, con el objetivo de asegurar la máxima objetividad y riqueza de perspectivas, de forma especial cuando se evalúa la calidad de vida de personas con autismo con discapacidad intelectual. La información obtenida debe orientarse hacia la mejora de las condiciones de vida, y los servicios y apoyos que recibe la persona con autismo.

A pesar de las dificultades de comunicación que presentan muchas personas con autismo y que justifican en gran medida el diseño de la

Guía de Indicadores, se propone complementar su implantación con el uso de otros instrumentos de evaluación de calidad de vida referidos a una dimensión más subjetiva y centrada en la propia percepción personal, que nos puedan ofrecer información adicional.

La aplicación de la Guía de Indicadores de Calidad de Vida la realizará, de forma consensuada, un Equipo Evaluador que irá verificando en la organización el grado de cumplimiento de cada uno de los indicadores. Para ello el equipo deberá observar y tener en cuenta no sólo las variables objetivas de la organización, sino también las subjetivas más relevantes que puedan influir y ayudar a valorar el nivel de calidad de vida que esta promueve. Las variables objetivas son aquellas condiciones de los contextos más o menos cercanos a la persona, que pueden repercutir en su calidad de vida (estructura y adaptación de contextos, uso de sistemas alternativos de comunicación, etc). Las variables subjetivas reflejan el grado de satisfacción o las percepciones personales que cada individuo tiene sobre su vida y que están determinadas en gran medida por sus valores, intereses, expectativas (posibilidades de elección, actividades adaptadas a los intereses, nivel de participación de las personas, evaluación de la satisfacción, etc).

La Guía de indicadores se complementa con una aplicación tecnológica que facilita el proceso de recogida e interpretación de datos, y la operatividad al realizar los procesos mecánicos relacionados con la aplicación del instrumento por parte del Equipo Evaluador (evaluación cuantitativa, elaboración de gráficos, etc).

Descripción de la guía de indicadores

La Guía de Indicadores de Calidad de Vida es un instrumento de evaluación de los factores contextuales referidos a las organizaciones y servicios de apoyo a las personas con autismo que inciden significativamente, de forma directa o indirecta, en su calidad de vida. Este instrumento consta de 70 indicadores agrupados en seis ámbitos, que definen los diferentes aspectos que deben ser tenidos en cuenta en una organización o servicio de apoyo para asegurar y evaluar su impacto en la calidad de vida de las personas:

1. **Calidad referida a la persona:** En este ámbito se valoran aspectos organizativos que tienen un impacto directo sobre la calidad de vida referida de la persona con autismo, atendiendo a cada una de las dimensiones propuestas por Schalock (1996): bienestar físico, bienestar

emocional, bienestar material, relaciones interpersonales, desarrollo personal, derechos, autodeterminación, inclusión social, sino en la de aquellas personas que conviven con ellas y que conforman sus contextos vitales más cercanos: familia y profesionales.

Entendemos que unas condiciones de vida saludables en familias y profesionales generan directamente, entre otros aspectos positivos, mejores condiciones para prestar apoyo y una mejor relación con la persona con autismo. Cuando una organización facilita la mejora de la calidad de vida de familias y profesionales reforzando vías de motivación, implicación y reconocimiento, está generando un impacto positivo en la vida de la persona con autismo.

2. **Identificación de las necesidades y preferencias / Elaboración y seguimiento de los planes de desarrollo personal:** Un proceso de detección de necesidades, planificación de metas y diseño de apoyos, debe realizarse de forma coordinada, implicando a todas aquellas personas significativas en la vida de la persona con autismo, y facilitar el que esta tenga un papel realmente activo, de forma que el plan individual de apoyo responda a sus intereses, capacidades y preferencias.
3. **Formación de profesionales:** A su perfil personal, el profesional debe sumar un amplio conocimiento del autismo y de cada persona con la que interviene, además del dominio de diferentes técnicas y metodologías que faciliten su intervención, la coordinación de apoyos en contextos diversos contextos y la adaptación a las necesidades e intereses de la persona.
4. **Estructura y organización:** En este ámbito se valoran aspectos referidos a los agrupamientos, organización del trabajo, horarios, comunicación/coordinación y análisis de situaciones susceptibles de mejora que pueden favorecer el bienestar de las personas con autismo.
5. **Recursos y servicios:** La respuesta a las necesidades de la persona con autismo requiere de una determinada provisión de recursos personales y materiales, y de su óptima organización.
6. **Relación con la comunidad / Proyección social:** La inclusión social es una de las dimensiones claves de la calidad de vida, y en este ámbito se valoran diferentes aspectos referidos a cómo la organización se proyecta hacia el exterior y facilita la participación en la comunidad de las personas con autismo.

La utilización de indicadores como medida de evaluación, tal como indican *Verdugo et al. (2006)*, es útil para mejorar resultados, puesto que su medida es significativa e interpretable, y permiten la recogida de datos sin excesivo esfuerzo y, como en el caso de esta Guía, están basados en un modelo teórico validado y decidido por consenso.

En esta línea se configura la Guía de Indicadores de Calidad de Vida, como un instrumento que pretende ser sensible a los apoyos y condiciones de las organizaciones relacionados con el bienestar de la persona.

Cada indicador consta de cuatro evidencias, es decir, cuatro realidades fácilmente observables que ayudan a hacer cuantificable el indicador, y a poder comprobar si este se cumple o no con un mismo criterio de valoración objetivo para todos los miembros del Equipo Evaluador.

Metodología de aplicación

Para la aplicación de la Guía de Indicadores de Calidad de Vida se deberá determinar:

- Equipo Evaluador.
- Planificación de la evaluación.
- Sesiones de trabajo.

Equipo Evaluador del Plan de Calidad de Vida

Por *Equipo Evaluador del Plan de Calidad de Vida (ECPCV)* se entiende el grupo de personas que evalúa los indicadores y si se cumplen o no las evidencias de cada uno de ellos. En caso de evaluar distintos servicios o centros de una misma organización, puede crearse un Equipo Evaluador diferente para cada uno de ellos. Es recomendable que los evaluadores sean multidisciplinares, potenciando así una visión integral de la organización. Para favorecer El Equipo Evaluador estará compuesto, al menos, por:

- **Evaluador principal:** Profesional externo a la organización donde se va a realizar la evaluación, con formación y experiencia en la aplicación de instrumentos relacionados con sistemas de gestión de calidad. Éste, será el encargado de dirigir el proceso de aplicación de los indicadores y contrastar cada uno de ellos definiendo también el papel que para esta tarea pueden desempeñar los demás componentes del Equipo Evaluador.

- **Responsable de la organización o del servicio evaluado:** Su función principal será servir de guía e intermediario entre el Evaluador Principal y la organización o el servicio, facilitando a éste el acceso a la información y a toda persona que pueda brindar evidencias que permitan contrastar los indicadores. Esto resulta especialmente necesario en el caso de grandes organizaciones, donde la persona responsable no conozca en profundidad todos los servicios. La persona responsable del servicio o la organización, además de aportar la información propia del cargo, ejercerá las funciones de “secretario”, tomando nota de la información recogida y de los acuerdos tomados.
- **Un familiar o persona significativa de una de las personas con autismo:** El familiar o la persona significativa en la vida de la persona con autismo debe conocer la organización y siempre que sea posible ser designado por ésta para representarla. Su papel se centrará en facilitar la evaluación, ayudando a encontrar evidencias que permitan comprobar cada uno de los indicadores.
- **Un profesional de atención directa:** Designado por la organización, su función consistirá en aportar y facilitar el acceso a la información, guiando la búsqueda de evidencias que ayuden a evaluar cada uno de los indicadores. Además del conocimiento de la organización o servicio, sería oportuno que este profesional tuviera conocimientos o experiencia en el ámbito de los sistemas de gestión de calidad.
- **Una persona con autismo:** Se facilitará, utilizando los apoyos que necesite, la participación de una persona con autismo. En los servicios a personas con mayores necesidades de apoyo, puede recurrirse a consultas realizadas a varias personas que, de forma directa o indirecta, puedan aportar información para valorar el grado de cumplimiento de los indicadores y evidencias, además de la favorecer la obtención de información por procedimientos indirectos (*Vidriales et al. (2017).*) Para aquellos centros o servicios más grandes, y de cara a enriquecer el proceso a través de integrar una visión más completa desde cada uno de los grupos de interés, cabe la posibilidad de que puedan participar más personas a juicio del evaluador principal y la dirección del centro o servicio. Si bien, siempre habrá de respetarse la proporción apuntada para cada grupo de interés, salvo en el caso del evaluador principal que, al ser externo, solo será una persona.

Planificación de la evaluación

Esta fase se iniciará con una visita previa a la organización, en la que todo el Equipo Evaluador tendrá la oportunidad de conocerse y planificar el proceso. El Evaluador Principal, acompañado del resto del equipo, conocerá así la organización, lo que le permitirá situarse para desarrollar la evaluación.

En esta visita previa se debe definirse el calendario de las posteriores sesiones de trabajo y planificar una estimación del tiempo necesario para la aplicación de la herramienta de evaluación. La experiencia desarrollada nos orienta a que la valoración se realice en las siguientes condiciones:

- Al menos tres reuniones, de dos horas cada una.
- Desarrollas en un periodo no superior a un mes.

Siempre que sea posible, el primer día de reunión se fijarán las fechas en las que se reunirá el equipo para realizar la evaluación.

Sesiones de trabajo

Las reuniones del Equipo Evaluador tendrán lugar en el centro en el que las personas con autismo desarrollan prioritariamente la actividad, o desde el que se lleva a cabo la planificación de apoyos y el seguimiento.

Una vez en la organización, el Equipo Evaluador tendrá la posibilidad de solicitar la presencia puntual de otros profesionales de referencia en los diferentes ámbitos que engloba el instrumento: formación, planificación, organización y recursos, etc, para solicitar información que les ayude a valorar cada uno de los indicadores a través de las evidencias.

Para encontrar evidencias podemos recurrir a distintas vías:

- Observación directa.
- Análisis de documentación. o Contacto e intercambio con los profesionales.
- Consulta directa o indirecta a personas con autismo

Para este proceso es importante que todas las personas participantes cuenten con una copia de la Guía de Indicadores, así como la Plantilla de Registro de Indicadores y Evidencias que les permita anotar la información.

Tabulación de los datos

A las personas										Identificación de necesidades	Formación de profesionales	Estructura y organización				Recursos	Comunidad
De interés fundamental	2	7				16				32	46				58		
	1	6	11			17				30	45					67	
De interés alto	5	10			15					28	34			50	53	55	
	3	8		13	14		18	20		31	35	37	40		54	56	
De interés medio									25		41						70
	4	9	12				19		21	24	33	38		49	52	57	69
De menor interés									23	28		42				60	
									22	27		39	47	48		59	
Perspectiva de las personas con autismo										Estructura y organización				Recursos			
Bienestar físico																	
Bienestar emocional																	
Bienestar material																	
Relaciones interpersonales																	
Desarrollo personal																	
Derechos																	
Autodeterminación																	
Inclusión social																	
Perspectiva de las familias																	
Perspectiva de los profesionales																	
Programación																	
Planificación de apoyos																	
Plan de seguimiento y evaluación																	
Conocimiento del autismo																	
Planificación centrada en la persona																	
Actitudes y valores																	
Participación en investigaciones																	
Grupos de iguales																	
Organización de la actividad																	
Horario																	
Comunicación/coordiación																	
Evaluación sistemática																	
Liderazgo																	
Innovación																	
Recursos																	
Comunidad																	

Figura 3.1: Tabulación de datos en evaluación completa

La calidad de vida es un concepto que, por un lado, se desarrolla a lo largo de un continuo, y como tal, escapa a los parámetros estadísticos del todo o nada, y por otro, se halla en un proceso permanente de mejora, por lo que los indicadores han de servir para ayudar a situar el alcance de dicho proceso y enfocar las actuaciones de mejora derivadas del mismo. La evaluación tiene carácter progresivo, identificando distintos tramos o grados de avance hacia la excelencia. Más allá de suponer un hito aislado o un reconocimiento obtenido en un momento determinado.

La información que se ha recogido dará lugar a dos tipos de información:

- Por una parte, se contará con un Gráfico de la organización o servicio que facilitará la información acerca de sus debilidades y fortalezas en relación a la calidad de vida.
- Por otra, obtendrá una puntuación global de la organización.













Nivel		Nº de indicadores anotados	Multiplicar por	
De interés fundamental			5	
			4	
			0	0
Interés alto			4	
			3	
			0	0
Interés medio			3	
			2	
			0	0
Menor interés			2	
			1	
			0	0
TOTAL				

Figura 3.2: Tabla de puntuaciones por puntuación y por colores

Puntuación	Nivel	Significado
200 - 250	Excelente	La organización o el servicio promueve un alto nivel de calidad de vida para las personas con autismo en todos los ámbitos, y asume la necesidad de desarrollar un proceso de mejora continua.
150 – 200	Muy bueno	La organización o el servicio promueve calidad de vida para las personas con autismo en todos sus ámbitos, aunque se observan algunas cuestiones organizativas que deberían corregirse. Se plantean algunas observaciones cuyo cumplimiento ayudarán a elevar el nivel.
100 – 150	Bueno	La organización o el servicio promueve calidad de vida para las personas con autismo, aunque se proponen algunas sugerencias de mejora que deben tenerse en cuenta poniendo especial atención en aquellas referidas a los indicadores de mayor interés.
50 – 100	Mejorable	Se requiere una nueva aplicación de la Guía que permita revisar de nuevo el cumplimiento de los indicadores no conseguidos. La organización o el servicio puede mejorar la calidad de vida que
		facilita a las personas con autismo, y para ello debe diseñar un Plan de Mejora y volver a aplicar nuevamente la Guía transcurrido el tiempo necesario para poder implantarlo.
0 – 50	Muy mejorable	Se requiere una nueva aplicación de toda la Guía. La organización o el servicio no tiene implantado el modelo de calidad de vida. Es fundamental un Plan de Mejora, su aplicación inmediata y la posterior reevaluación.

Figura 3.3: Tabla de rangos de puntuación total

Por tanto, el primer paso para nuestro sistema experto será el siguiente, determinar el nivel del indicador a partir del número de evidencias:

- Si se ha marcado una evidencia o no se ha marcado ninguna, el estado será *En comienzo* y por tanto se coloreará de color rojo.
- Si se han marcado dos o tres evidencias, el estado del indicador será *En proceso* y por tanto se coloreará de color amarillo.
- Si se han marcado todas las evidencias, el estado del indicador será *Conseguido* y por tanto se coloreará de color verde.

Y a partir de esos resultados:

- Si se colorea de rojo o amarillo, sí que debe estar en el plan de mejora.
- En caso de que se coloree en verde, no debe estar en el plan de mejora.

Esta conclusión del proceso la deberá realizar el Equipo Evaluador, encargado también de elaborar el Informe Final, que servirá de referencia para la elaboración posterior del Plan de Mejora por parte de los responsables de la organización o servicio.

El Plan de Mejora debe ser presentado al Equipo Evaluador para que realice las aportaciones que estime oportunas, y posteriormente sea presentado y consensuado con el resto de miembros de la organización o del servicio. Una vez definido y consensuado el Plan de Mejora, el Equipo Evaluador planificará una nueva evaluación de la organización o el servicio, fijándose como referencia un plazo de 2 o 3 años, que coincidirá con el periodo de implementación del Plan de Mejora aprobado.

Los resultados que se pueden obtener son los siguientes:

Puntuación	Nivel
200-250	Excelente
150-200	Muy bueno
100-150	Bueno
50-100	Mejorable
0-50	Muy mejorable

Figura 3.4: Rangos de puntuación en evaluación completa

Guía de indicadores

Los indicadores que componen esta guía compuesta, en acompañamiento de sus cuatro evidencias, son los siguientes:

- **Calidad referida a la persona:**
 - **Calidad desde la perspectiva de la persona con autismo:**
 - **Bienestar físico:**
 - ◊ ***Indicador 1: Existen programas de atención a la salud personalizados y actualizados***
Evidencias:
 1. Se dispone de un expediente de salud individual, confidencial y actualizado, que contiene información referida a: historial, comorbilidad, medicación, pruebas realizadas, necesidades referidas a la alimentación, actividad física, etc.
 2. Se realizan revisiones periódicas de prevención y seguimiento de la salud, incluyendo las recomendadas en función de la edad y condiciones específicas de salud (dificultades de visión, de audición, etc.).
 3. En los casos necesarios se desarrollan acciones de sensibilización, adaptación a los entornos sanitarios y a las diferentes pruebas médicas.

4. La persona tiene acceso a un cuadro médico de especialistas estables y conocedores de las características del autismo, de distintas especialidades: medicina general, psiquiatría, odontología, ginecología, etc.

◇ ***Indicador 2: Se garantiza la correcta administración y seguimiento de los tratamientos de salud.***

Evidencias:

1. Existe un protocolo-proceso de la administración, cuando esta deba ser administrada en el contexto del servicio (responsable, control, autorizaciones, etc).
2. Existen registros que garantizan la correcta administración de la medicación y reflejan posibles incidencias.
3. Se registran, analizan y se informa a la familia y a los médicos que lo han prescrito, de los posibles efectos secundarios observados derivados de los cambios de medicación.
4. Se realizan análisis periódicos de control y seguimiento de las medicaciones.

◇ ***Indicador 3: Se interviene de manera personalizada en el ámbito del cuidado y promoción de la autonomía personal.***

Evidencias:

1. Existen planes individuales de apoyo que contienen objetivos referidos a la promoción de la autonomía personal, que se evalúan y actualizan de manera periódica.
2. Los objetivos de promoción de la autonomía personal se trabajan en los contextos naturales de las actividades de la vida diaria (alimentación, vestido, aseo, etc.).
3. Los profesionales que trabajan en contacto directo con las personas con autismo conocen y coordinan las pautas a seguir para procurar el bienestar físico de cada una de ellas, a través de la promoción de habilidades referidas a la autonomía personal: vestido, higiene, comida, autonomía personal, etc.
4. La persona dispone de condiciones adecuadas, espacios y tiempos de privacidad para el desarrollo de las actividades de cuidado y autonomía.

◇ ***Indicador 4: Se desarrollan actuaciones referidas a la seguridad e higiene en los diferentes***

entornos en los que se desarrolla el apoyo a las personas.

Evidencias:

1. Existe un plan de identificación individualizada de los riesgos (situaciones, materiales, actividades, etc.) referidos a cada persona.
 2. Se dispone de un plan de gestión del equipamiento que garantiza la formación y el uso adecuado de los productos de apoyo y dispositivos de seguridad que, sin crear un entorno restrictivo, favorecen la autonomía y la seguridad de las personas, minimizando los riesgos que pueden provenir del contexto.
 3. Las instalaciones, productos, bienes y servicios facilitan la comprensión y el desenvolvimiento autónomo por parte de la persona, garantizando la ausencia de barreras cognitivas, físicas o sensoriales.
 4. Se dispone de protocolos de formación e intervención que permiten prevenir y abordar situaciones de emergencia de forma eficaz (planes de buen trato, planes de prevención de abusos, planes de evacuación, primeros auxilios, plan de prevención de riesgos laborales, etc.).
- ◇ ***Indicador 5: Se contemplan medidas preventivas personalizadas para mantener una salud adecuada.***

Evidencias:

1. Las condiciones físicas, sensoriales y requerimientos cognitivos para el desenvolvimiento de la persona con autismo se adaptan a sus necesidades ergonómicas promoviendo su bienestar físico: adecuada luz y temperatura, control postural, ruido ambiental...
2. Se promueve una nutrición adecuada, que además incorpora las preferencias y gustos personales (menús adaptados, dietas, adecuación a las posibilidades de deglución de cada persona, seguimiento de hábitos alimentarios, etc.).
3. Cada persona participa en programas dirigidos a mantener una vida saludable y prevenir un posible deterioro físico (control de peso, ejercicio físico, deporte, fisioterapia, higiene, prevención de Trastornos de Conducta Alimentaria, conocimiento de riesgos de salud más pre-

valentes en mujeres con autismo...), con profesionales especializados.

4. La persona participa en programas de formación y promoción de la salud, referidos, entre otros, a la autonomía personal, la prevención de adicciones, la sexualidad y afectividad.

○ **Bienestar emocional:**

- ◇ ***Indicador 6: La persona se desenvuelve en un contexto accesible, comprensible y seguro, que minimiza el estrés.***

Evidencias:

1. Existen condiciones de estructuración en el contexto (espacio, actividades, horarios, etc.) que favorecen un entorno predecible.
2. Se interviene de forma personalizada en los problemas emocionales.
3. Cualquier nueva intervención o tratamiento por parte de los profesionales, se pone en práctica tras obtener el consentimiento informado de la persona con autismo o en su caso de las personas que les apoyan en la toma de decisiones.
4. Existe una identificación de afinidades y preferencias personales que se incorpora al plan individualizado de apoyos y se tiene en cuenta para configurar los sistemas de apoyo (grupo de participantes, actividades propuestas, etc.).

- ◇ ***Indicador 7: Se promueve el máximo bienestar emocional en la vida de la persona con autismo.***

Evidencias:

1. Existe un plan de actuación personalizado que permite prever y abordar los riesgos que pueden comprometer el bienestar emocional de la persona (situaciones imprevistas, cambios en horario o actividades, ausencia de profesionales de referencia, etc.).
2. Existe una estructura flexible de funcionamiento que permite resolver de forma inmediata los imprevistos que afectan a la estabilidad en la organización: ausencia de un profesional, cambios en las actividades previstas, alteración de espacios, etc.

3. Se utilizan sistemas de información y estructuración ambiental y temporal que facilitan la orientación y el uso de los distintos espacios, favoreciendo la accesibilidad y los principios de diseño universal.
4. La persona con autismo tiene personas de referencia claras en su vida (familiares, profesionales, iguales, amigos), pudiendo contar con referentes en situaciones de urgencia (procesos de duelo, desregulación emocional, tiempos con menor estructura, interrupción temporal de atención profesional, etc.).

◇ ***Indicador 8: Se desarrollan programas personalizados basados en el apoyo conductual positivo.***

Evidencias:

1. Existen unas pautas generales de prevención de conductas problemáticas (guía de convivencia y funcionamiento, reglamento de régimen interno, plan de atención libre de sujeciones, etc.).
2. Existe un registro de las dificultades conductuales y de autorregulación, y un análisis funcional dirigido a identificar los factores que las originan o mantienen.
3. Se realizan cambios en la organización o en las rutinas, enfocados a prevenir las conductas problemáticas.
4. Existe un plan de capacitación y abordaje de las dificultades conductuales y de autorregulación que establece las medidas de prevención o intervención (desarrollo de habilidades, modificación de contextos y rutinas, implantación de apoyos, etc.), basadas en parámetros éticos y respetuosos con la dignidad y los derechos de la persona.

◇ ***Indicador 9: La persona con autismo participa en la planificación, ejecución y evaluación de su Plan Individual de Apoyos.***

Evidencias:

1. Existen canales de expresión y participación de las personas con autismo en relación con el Plan Individual de Apoyos.
2. La persona toma decisiones sobre su vida, disponiendo de los apoyos personalizados que requiera para ello.

3. Las actividades se adaptan y estructuran de forma que se garantiza el éxito en su realización de la forma más autónoma posible.
4. La persona participa en la evaluación y actualización de su plan individualizado de apoyos, disponiendo de los apoyos que requiera para expresar sus preferencias y tomar decisiones.

◇ ***Indicador 10: Las personas con autismo cuentan con apoyos personalizados.***

Evidencias:

1. La persona dispone de una o varias figuras profesionales de referencia que participan de manera más activa en su plan de apoyos, con las que mantiene una mayor afinidad e implicación.
2. La persona comparte un círculo de iguales con los que tiene una mayor afinidad atendiendo a criterios de género, edad, intereses, personalidad, etc., y con los que puede compartir tiempo y actividades de manera cotidiana.
3. La organización cuenta con el personal necesario para apoyar a las personas en el entorno comunitario.
4. Se promueve y apoya a las personas del entorno para que se impliquen como apoyos naturales de las personas con autismo.

○ **Bienestar material:**

◇ ***Indicador 11: Se respeta la intimidad.***

Evidencias:

1. La persona con autismo dispone de espacios, tiempos y pertenencias personalizadas y significativas, garantizándose la intimidad en su uso y disfrute.
2. La persona toma decisiones sobre el uso o el acceso a sus espacios y pertenencias personales.
3. Se favorece la intimidad en la realización de actividades referidas al aseo, vestido, cuidado personal, teniendo en cuenta la diversidad sexual, promoviendo la preservación de los derechos y el trato acorde con la edad.
4. El uso de la imagen e información sobre las personas con autismo está sujeto a un protocolo de utilización y

a la normativa sobre protección de datos que garantiza el respeto y la confidencialidad.

◇ ***Indicador 12: Se promueve la disponibilidad, cuidado y acceso a pertenencias personales.***

Evidencias:

1. La persona dispone de pertenencias y recursos ajustados y suficientes para responder a sus necesidades personales básicas (ropa, calzado, medicamentos, etc.).
2. La persona dispone de espacios estables y pertenencias personales cuidadas, adecuadas a la edad cronológica, y ajustadas a las preferencias personales.
3. Cada persona recibe un refuerzo o contraprestación por su actividad.
4. La persona gestiona sus recursos, dinero y pertenencias personales, contando con los apoyos que pueda necesitar.

○ **Relaciones interpersonales:**

◇ ***Indicador 13: Se promueven las relaciones sociales significativas y las competencias necesarias para su disfrute.***

Evidencias:

1. La persona expresa preferencias, y se tienen en cuenta para configurar los grupos de participantes de las actividades que realiza.
2. Existen programas personalizados de inclusión social y laboral, que fomentan la interacción con iguales y personas sin discapacidad en los diferentes entornos.
3. El plan individualizado de apoyos incorpora objetivos relacionados con la promoción de las competencias de comunicación social y el establecimiento de relaciones personales significativas.
4. Se promueve y apoya a las personas del entorno para que se impliquen como apoyos naturales de las personas con autismo.

○ **Desarrollo personal:**

◇ ***Indicador 14: Se promueve el desarrollo de las capacidades e intereses individuales.***

Evidencias:

1. La persona dispone de un plan individualizado de apoyos basado en una evaluación de su calidad de vida, objetiva y subjetiva, competencias, necesidades, intereses y preferencias individuales.
2. El plan individualizado de apoyos se revisa periódicamente y se actualiza conforme a los resultados obtenidos o a las nuevas necesidades identificadas.
3. Las actividades que se desarrollan en los distintos programas o servicios, se diseñan o seleccionan de forma que, además de dar respuesta a las necesidades, intereses y capacidades personales, responden a un criterio de funcionalidad.
4. La persona accede a distintos itinerarios personalizados de apoyo que fomentan su desarrollo personal y el alcance de metas significativas en distintas dimensiones de su vida (educación, formación, empleo, inclusión social, vida independiente, etc.).

◇ ***Indicador 15: Se promueve el avance y el desarrollo continuo de la persona en diferentes ámbitos de la vida (formación, ocio, laboral, etc.).***

Evidencias:

1. La persona dispone de apoyos y ajustes personalizados para acceder a contenidos y actividades que promuevan su desarrollo personal.
2. La persona accede a programas formativos, actividades y materiales de aprendizaje que resultan acordes a su edad, necesidades y capacidades, y responden a criterios de accesibilidad universal en su diseño y usabilidad.
3. Existe un sistema de evaluación continua del desarrollo personal, que realiza un seguimiento del progreso y de las barreras/facilitadores que inciden en el mismo, y favorece el ajuste de los sistemas de apoyo personalizados que requiere la persona (tipología, frecuencia, intensidad, etc.).
4. Se planifica y promueve la retirada gradual de apoyos.

○ **Derechos:**

◇ ***Indicador 16: Se garantiza el respeto a la identidad y dignidad de la persona.***

Evidencias:

1. La organización cuenta con normas de funcionamiento interno, accesibles, que aseguran los derechos y deberes referidos a profesionales, familias y personas con autismo, que tienen como referencia la Declaración Universal de los Derechos Humanos y, de forma especial, la Convención Internacional sobre las Personas con Discapacidad.
2. La persona participa en la elaboración de los criterios éticos que deben guiar la facilitación de los apoyos que necesita o desea.
3. La persona, y/o quienes facilitan apoyo a la persona con autismo, conocen sus derechos fundamentales con relación a los apoyos que estas precisan y desean, y los ejercen con garantías.
4. La organización no es restrictiva, fomenta nuevas oportunidades, no coarta las posibilidades de elección ni de desarrollo de las personas a las que apoya.

◇ ***Indicador 17: Se garantiza la integridad física.***
Evidencias:

1. No se utiliza ningún tipo de restricción física, ni tratamiento farmacológico si no está previamente justificado y consensuado con la familia, un técnico cualificado y, en los casos en que sea posible, directamente con la persona con autismo, y en caso de fármacos éstos deben haber sido siempre prescritos desde el ámbito médico.
2. La persona dispone de medidas de apoyo que minimizan los riesgos en la realización de las actividades de la vida cotidiana (anticipación, práctica previa, resolución de problemas, etc.).
3. Existen pautas y medidas para prevenir o extinguir riesgos físicos derivados de conductas problemáticas, medidas punitivas, abusos físicos, emocionales, sexuales (hojas de quejas, planes de contingencia ante emergencias, protocolo de prevención, detección y denuncia de situaciones de malos tratos, y medidas de promoción del buen trato).
4. La persona dispone de recursos de apoyo, ayudas y adaptaciones técnicas que garantizan el acceso, com-

prensión y uso seguro de espacios, productos, bienes y servicios.

○ **Autodeterminación:**

◇ ***Indicador 18: Las personas expresan opiniones, preferencias y toman decisiones significativas sobre sus vidas.***

Evidencias:

1. Las personas reciben formación variada y adaptada, previa a la emisión de conductas de autodeterminación.
2. Se apoya el que la persona comprenda y planifique la secuencia de pasos de las actividades que realiza.
3. La persona accede a la información que requiere para la expresión de opiniones y la toma de decisiones en formatos cognitivamente accesibles, que se ajustan a sus capacidades y necesidades.
4. La persona dispone de oportunidades para tomar decisiones de manera frecuente y cotidiana.

◇ ***Indicador 19: Las personas con autismo participan en el diseño, implementación y evaluación de sus Planes Individuales de Apoyo.***

Evidencias:

1. Se anticipan y planifican las actividades a realizar, disponiendo toda persona de momentos en los que puede elegir libremente qué hacer o no hacer en su tiempo libre, promoviendo el conocimiento de las distintas posibilidades de ocio.
2. La persona tiene oportunidades y dispone de un sistema de comunicación adaptado, para comunicar necesidades, emociones, y para realizar elecciones.
3. Se promueven actividades encaminadas a desarrollar capacidades de planificación que permitan a la persona elegir o participar en las decisiones que afectan a su vida, tanto en lo referido a cuestiones cotidianas como a cuestiones de mayor trascendencia para su futuro.
4. Se aprovechan o provocan situaciones controladas o riesgos asumibles (posibles imprevistos, situaciones en las que es necesario pedir ayuda...), que favorecen el desarrollo de habilidades de resolución de problemas en distintos contextos cotidianos, se trabajan estrategias

y se ofrecen apoyos a la persona para su resolución o afrontamiento.

◦ **Inclusión social:**

◊ ***Indicador 20: Se promueve la inclusión social de las personas con autismo.***

Evidencias:

1. Se realiza un análisis ecológico y funcional previo a la inclusión social de la persona.
2. La persona participa en actividades y programas realizados en distintos entornos comunitarios, favoreciendo, situaciones de inclusión inversa.
3. Se promueve el desarrollo de objetivos sociales y comunicativos en contextos naturales, atendiendo a las necesidades de la persona en el entorno próximo a su domicilio, de forma que faciliten su participación y relación con recursos, servicios y otras personas de su vecindario.
4. Se utilizan los medios de comunicación convencionales y redes sociales para la información y divulgación hacia la sociedad.

• **Calidad desde la perspectiva de las familias:**

◦ ***Indicador 21: Las actuaciones con la persona con autismo tienen en cuenta a la familia, en los casos que sea pertinente.***

Evidencias:

1. El Plan Individual de Apoyos integra las expectativas de la familia que se ajustan a las necesidades y capacidades de las personas con autismo.
2. Existen procedimientos para recoger y revisar periódicamente las expectativas de la familia hacia la organización.
3. La organización promueve el que la familia se integre activamente en la red de apoyos de la persona con autismo.
4. Los objetivos del Plan Individual de Apoyos respetan el estilo de vida y de relación familiar de la persona con autismo.

◦ ***Indicador 22: Se facilita la implicación de las familias en la organización, en los casos que sea pertinente.***

Evidencias:

1. La familia participa en la elaboración del Plan Individual de Apoyos de la persona con autismo, siempre que la situación lo requiera, y puede tener información de su evolución en cualquier momento.
2. Existe un plan de formación y asesoramiento a las familias, con profesionales que las conocen.
3. Existe una variedad de vías de implicación y participación en la organización.
4. Existe un sistema de comunicación / coordinación permanente con los servicios y profesionales, que garantice el seguimiento continuado.

◦ ***Indicador 23: Se favorece un aumento del nivel de satisfacción en las familias.***

Evidencias:

1. Existen vías para medir y analizar el nivel de satisfacción: encuestas, entrevistas personales, sistema de quejas, sugerencias, recepción de felicitaciones, etc.
2. Existen vías para comunicar incidencias que puedan alterar la convivencia normalizada, realizar sugerencias y propuestas de mejora, informar de intervenciones específicas.
3. Se analizan y tienen en cuenta las incidencias y propuestas formuladas por las familias.
4. Se implica a las familias en los procesos de mejora, garantizando que reciben información adecuada y suficiente.

• **Calidad desde la perspectiva de los profesionales:**

◦ ***Indicador 24: Se conocen, valoran y se tienen en cuenta las propuestas e iniciativas provenientes de los profesionales.***

Evidencias:

1. Existe fácil acceso por parte de los profesionales hacia el equipo directivo.
2. Se registran y valoran las propuestas de intervención/organización, provenientes de los profesionales.
3. Existen planes de desarrollo profesional, ajustados a las expectativas individuales, complementados con un plan de formación permanente.
4. Se solicitan aportaciones de los profesionales, referidas a distintos proyectos de la organización.

- ***Indicador 25: Las responsabilidades de los profesionales son coherentes con sus niveles de competencias.***

Evidencias:

1. Existe un organigrama que especifica la estructura del personal de la organización, así como las competencias asociadas a cada puesto.
2. Cada profesional conoce sus responsabilidades.
3. Las funciones, responsabilidades y competencias son ajustadas al puesto de trabajo.
4. Existe una política retributiva transparente y justa, así como de promoción y desarrollo profesional.

- ***Indicador 26: Se promueve la participación y el trabajo en equipo.***

Evidencias:

1. La estructura de la organización contempla el funcionamiento a través de equipos de trabajo para el desarrollo de proyectos y desafíos.
2. Existen grupos de mejora e innovación para evaluar periódicamente y hacer propuestas de mejora en la organización.
3. Existen oportunidades de abordar en grupo las estrategias puntuales de intervención, o de apoyo a la intervención, planteadas por cualquier profesional.
4. La dirección de la organización promueve la estabilidad del equipo de profesionales, desarrollando una política de retención del talento.

- ***Indicador 27: Se promueve la mejora del nivel de satisfacción en los profesionales.***

Evidencias:

1. Existen vías para medir y analizar el nivel de satisfacción y motivación profesional: encuestas, entrevistas personales, etc.
2. Se reconocen, valoran y difunden las buenas prácticas profesionales desarrolladas en la organización.
3. La organización valora y promueve las cuestiones que inciden de forma específica en la satisfacción y motivación de cada profesional.

4. Se planifican, recogiendo sugerencias de los profesionales, estrategias que inciden directamente en su satisfacción y bienestar (planes de prevención de riesgos psicosociales, prevención y actuación en casos de burnout, planes de movilidad, planes de acogida del personal, facilidades en la conciliación de la vida familiar con la laboral, etc.) y se evalúan.

○ ***Indicador 28: Los profesionales están implicados en la organización.***

Evidencias:

1. Se implica a todo el personal en los procesos de mejora.
2. Los profesionales participan en la toma de decisiones organizativas y/o de planificación.
3. Existe información sobre los proyectos de la organización y sobre los resultados que esta consigue en diferentes ámbitos.
4. Se favorece la implicación de los profesionales en los proyectos de la organización.

■ **Identificación de las necesidades y preferencias / elaboración y seguimiento de los planes individuales de apoyo:**

• **Planificación:**

○ ***Indicador 29: Se evalúan las necesidades, deseos y expectativas de las personas con autismo en los distintos ámbitos de intervención.***

Evidencias:

1. Existe un sistema de recogida de información sobre las necesidades y expectativas de la persona con autismo en los distintos ámbitos de intervención.
2. Para la elaboración de cada Plan Individual de Apoyos y la Programación General del Programa o Servicio, se analizan el conjunto de datos recogidos sobre cada persona con autismo.
3. En la valoración de las necesidades y diseño de planes, participan todos los profesionales y se implica a la familia siempre que se considere pertinente.
4. En la valoración de las necesidades se implica a la persona con autismo a través de distintas modalidades comunicativas o apoyos personalizados.

- ***Indicador 30: Los planes de apoyo se adaptan a las necesidades específicas a lo largo de toda la vida.***

Evidencias:

1. Los Planes Individuales de Apoyo incluyen la consecución de metas personales definidas tras la evaluación de calidad de vida de la persona con autismo.
2. Los Planes Individuales de Apoyo se consensúan entre todos los profesionales que están en contacto con la persona con autismo, con su familia, y con la persona con autismo.
3. Las personas implicadas en prestar apoyo a las personas con autismo están coordinadas en el uso de pautas específicas de intervención en diferentes ámbitos (conducta, rehabilitación funcional motora, corrección postural, comunicación, etc.).
4. Las actividades se adaptan y estructuran de forma que se garantiza el éxito y su realización de la forma más autónoma posible.

- ***Indicador 31: La estructura de la Programación General de la organización o del servicio se adapta a las características de las personas con autismo.***

Evidencias:

1. Existe una Programación General que engloba todos los ámbitos de intervención y que sirve de referente para realizar los Planes Individuales de Apoyo.
2. Los contenidos de la Programación General se evalúan periódicamente, y se modifican si se considera necesario.
3. Los objetivos de trabajo que promueve cada Plan Individual de Apoyo son concretos y medibles.
4. Existe un análisis que evidencia la funcionalidad de los objetivos y aprendizajes conseguidos.

- ***Indicador 32: El proceso de elaboración de los Planes Individuales de Apoyo se adecúa a las características de cada persona con autismo.***

Evidencias:

1. Existe un proceso de elaboración de los Planes Individuales de Apoyo en el que participan todos los profesionales que están en contacto con la persona con autismo, la

familia, y la persona con autismo, a través de distintas vías o estrategias de apoyo.

2. El Plan Individual de Apoyo detalla los objetivos, metas y los apoyos necesarios para su consecución.
3. Se realizan revisiones periódicas tanto de los objetivos como de las necesidades de apoyo de cada persona con autismo.
4. Existe flexibilidad y posibilidad de introducir nuevos objetivos o metas y/o modificar el tipo o grado de apoyo, cuando el Plan Individual de Apoyo ya está en marcha.

- **Planificación de apoyos:**

- ***Indicador 33: Los profesionales son una referencia clara para las personas con autismo.***

Evidencias:

1. Cada persona con autismo dispone de un profesional-tutor de referencia, contemplando la variación a lo largo del tiempo para evitar la excesiva dependencia emocional y la inercia de la rutina.
2. Existen unos criterios de asignación de los profesionales a la persona con autismo.
3. Se analiza periódicamente la relación y la adecuación del perfil humano y profesional a las características y preferencias de la persona con autismo, existiendo posibilidad de cambio de profesional de referencia.
4. Los profesionales-tutores canalizan toda la información pertinente sobre la persona con autismo, y coordinan las intervenciones y la prestación de apoyos en los distintos contextos.

- ***Indicador 34: Los apoyos se planifican de acuerdo a las necesidades de la persona.***

Evidencias:

1. La planificación de apoyos tiene en cuenta las aportaciones de la persona con autismo y de las personas de referencia en sus diferentes contextos vitales (familia, profesionales, amigos, conocidos, etc.).
2. Se captan y utilizan apoyos naturales.
3. Los apoyos permiten que la persona pueda conseguir sus objetivos y metas en los distintos ámbitos y contextos vitales.

4. Se dota a la persona de estrategias y habilidades que le permitan ejercer cambios y un control del entorno (elecciones, expresión de necesidades, resolución de problemas, etc.).

○ ***Indicador 35: Los criterios metodológicos se adaptan a las necesidades y capacidades de la persona con autismo.***

Evidencias:

1. Se utilizan estrategias y técnicas de intervención validadas, unificadas y compartidas en los distintos servicios, programas y contextos en los que participa la persona.
2. La definición y especificación de objetivos o metas personales facilita la interpretación objetiva, tanto en su ejecución como en su evaluación por parte de todos los profesionales.
3. Se contempla la generalización de aprendizajes.
4. Los objetivos o metas permiten planificar nuevos aprendizajes.

○ ***Indicador 36: Los Planes Individuales de Apoyo se adaptan a la persona.***

Evidencias:

1. El Plan Individual de Apoyo aborda todas las necesidades en las diferentes áreas de desarrollo personal y social.
2. El Plan Individual de Apoyo promueve el que la persona con autismo participe y realice actividades teniendo en cuenta variables como el género, intereses y capacidades, en distintos contextos, favoreciendo siempre la mayor inclusión posible.
3. Los Planes Individuales de Apoyos contemplan una amplia diversidad de opciones, adecuadas a los diferentes niveles de adaptación y capacidades.
4. Existen programas en función de las distintas etapas evolutivas (infancia, adolescencia, vida adulta, envejecimiento).

• **Plan de seguimiento y evaluación:**

○ ***Indicador 37: Se realiza un seguimiento y evaluación continua de cada Plan Individual de Apoyo.***

Evidencias:

1. Existen informes de evaluación individual de cada persona con autismo.
2. Se desarrollan procesos de evaluación de calidad de vida que miden el impacto de la intervención en la vida de las personas con autismo.
3. Se realizan orientaciones y propuestas de intervención futura basadas en la evaluación.
4. La información para realizar la evaluación y el seguimiento es aportada por la persona con autismo y las personas significativas en su vida.

■ **Formación de los profesionales:**

• **Conocimiento del autismo:**

- ***Indicador 38: Se asegura una formación inicial a los nuevos profesionales.***

Evidencias:

1. Existe un procedimiento de información, formación y apoyo a nuevos profesionales, personas que prestan apoyo natural, voluntarios, etc.
2. Se organizan acciones formativas en las que participan los nuevos profesionales, voluntarios, etc.
3. Existe una documentación formativa inicial que contiene información sobre autismo, programas y pautas de intervención.
4. Cada nuevo profesional tiene asignado un profesional-tutor que se responsabiliza de su formación y seguimiento, y una ficha personal de formación donde se recoge toda su trayectoria formativa y profesional.

- ***Indicador 39: La formación incluye aspectos técnicos, organizacionales y valores de la organización.***

Evidencias:

1. Existe un documento que define la naturaleza, finalidad, valores, objetivos y principios éticos de la organización.
2. Todos los profesionales tienen acceso a la política de la organización, los planes de mejora y a los procesos o procedimientos de gestión de calidad que les competen.
3. Existen canales de información para transmitir los fines, valores y objetivos de la organización.

4. Se ofrece información periódica sobre los proyectos y trayectoria de la organización.

○ ***Indicador 40: Cada profesional recibe formación específica sobre su puesto de trabajo.***

Evidencias:

1. Todos los profesionales en su proceso de formación inicial reciben una formación específica referida a su puesto de trabajo, en la que se promueve la participación activa y el aprendizaje significativo
2. El plan de formación específica incluye acciones para puestos de trabajo concretos.
3. Todos los profesionales tienen posibilidades, de forma periódica, de actualizar o ampliar la formación referida a su puesto de trabajo.
4. La organización promueve el que se realicen proyectos o iniciativas encaminadas a la formación específica o especialización: intercambios profesionales, grupos de trabajo específicos, etc.

○ ***Indicador 41: Se promueve la formación continua, la actualización y el desarrollo profesional.***

Evidencias:

1. Existe un plan de formación anual en el que se incluyen necesidades y demandas planteadas por los profesionales.
2. La organización colabora con otras entidades en proyectos de investigación y avance científico.
3. Todos los profesionales participan de forma periódica en acciones de formación realizadas por la entidad o por otras organizaciones.
4. Existe un procedimiento que evalúa periódicamente el desempeño de los profesionales.

○ ***Indicador 42: La entidad cuenta con recursos que favorecen la formación, actualización y desarrollo profesional.***

Evidencias:

1. Existe un programa de formación interna (cursos, reuniones de formación, intercambios con otras organizaciones, etc.).

2. La organización posibilita el acceso a vías de actualización: nuevas publicaciones específicas, investigación, bibliografía actualizada, Internet, etc.
3. Existe un sistema de gestión del conocimiento por el que toda la formación e información recibida por un profesional se difunde y se hace accesible al resto del equipo.
4. La organización mide la eficacia y el impacto de la formación profesional, recogiendo indicios de cómo esta produce cambios organizativos.

- **Conocimiento y adaptación a la persona con autismo:**

- ***Indicador 43: La intervención de cada profesional se adapta a las necesidades y características de cada persona con autismo.***

Evidencias:

1. El profesional tiene un conocimiento profundo de la persona con autismo, sus expectativas, metas. . . en los distintos contextos vitales.
2. El profesional impulsa y coordina la creación y desarrollo de grupos de apoyo que puedan implicar al resto de profesionales, familias, amigos. . . para definir y poner en marcha el Plan Individual de Apoyos.
3. El profesional conoce y aplica los apoyos necesarios para que la persona con autismo pueda participar en el diseño, desarrollo y evaluación de su Plan Individual de Apoyo, y para obtener información en relacionada con su calidad de vida.
4. El profesional promueve la captación de apoyos, especialmente naturales, para facilitar el Plan Individual de Apoyos.

- ***Indicador 44: Existe una información individualizada de cada persona con autismo.***

Evidencias:

1. Se utilizan instrumentos que permiten obtener datos significativos para la elaboración de cada perfil personal, incidiendo en preferencias, necesidades, visión de futuro. . .

2. Existe documentación personalizada (pautas específicas, gustos e intereses, capacidades, mapa de relaciones sociales, ficha personal, etc.).
 3. Existe un proceso que facilita el conocimiento de la persona con autismo y que contempla, tanto la información que ésta nos aporta sobre su vida y expectativas de futuro.
 4. Existen registros personales de seguimiento de las habilidades y metas que se trabajan con cada persona.
- ***Indicador 45: Se conoce en profundidad y de forma integral a la persona con autismo.***

Evidencias:

1. Los profesionales conocen la información referida a cada persona con autismo (pautas de intervención, gustos e intereses, nivel de capacidades...), y participan en el proceso de revisión y actualización de la misma.
 2. La información se refiere a todas las áreas y ámbitos vitales de la persona.
 3. Se utilizan instrumentos que permiten obtener datos significativos para la elaboración de cada Plan Individual de Apoyo, incidiendo en preferencias, necesidades, visión de futuro, etc.
 4. Se obtiene información directa de la persona con autismo y de las de personas cercanas que comparten experiencias vitales con ella: familias, profesionales, conocidos...
- ***Indicador 46: La intervención se adapta de forma personalizada en cada Plan Individual de Apoyos.***

Evidencias:

1. Los objetivos que se plantean con cada persona con autismo son funcionales y tienen un impacto real y significativo en su vida.
2. Las posibilidades de elección que tienen las personas con autismo tienen en cuenta sus gustos, intereses, visión de futuro, etc.
3. Las actividades, interacción y materiales están adaptados al perfil y edad cronológica de la persona con autismo.
4. Se tienen en cuenta las necesidades individuales de apoyo específico y/o especializado.

- **Actitudes y valores:**

- ***Indicador 47: La práctica y actitudes profesionales tienen como referente la misión y los valores de la organización o servicio.***

Evidencias:

1. La organización cuenta con un código ético que concreta su misión/visión y define los valores y principios que deben presidir la práctica profesional.
2. Todos los profesionales conocen la misión y los valores de la organización.
3. Todos los profesionales aceptan y se comprometen a regular su práctica basándose en el Código Ético.
4. Existe un sistema de reflexión que persigue integrar en la práctica profesional los valores y actitudes consensuados en el código ético y evaluar periódicamente en qué medida esto se cumple.

- **Participación en investigaciones sobre autismo:**

- ***Indicador 48: La organización promueve la ampliación del conocimiento a través de la participación activa en investigaciones sobre autismo.***

Evidencias:

1. Se mantienen convenios de colaboración con instituciones universitarias y científicas que realizan investigación sobre autismo.
2. La organización implica a las personas con autismo en investigaciones sobre autismo basadas en protocolos definidos en guías de buenas prácticas.
3. La organización colabora en investigaciones sobre autismo.
4. Se aplican o incorporan en la práctica los resultados de investigaciones científicamente contrastadas.

- **Estructura y organización:**

- **Grupos de iguales:**

- ***Indicador 49: La configuración de los grupos de iguales se adapta a las necesidades de las personas con autismo.***

Evidencias:

1. Existen criterios que justifican los grupos de iguales en los que se incluyen las personas con autismo: edad, capacidades, sexo, preferencias...
2. Se analiza periódicamente la interacción entre los componentes de los grupos de iguales.
3. Se detectan y resuelven los posibles conflictos y/o incompatibilidades detectadas.
4. Existe flexibilidad favoreciendo la elección, para realizar nuevos grupos de iguales ante situaciones o actividades puntuales o imprevistas.

- **Organización de la actividad:**

- ***Indicador 50: Las personas con autismo tienen una organización clara y accesible de las tareas y actividades en las que participan.***

Evidencias:

1. Existe una amplia variedad de tareas y actividades que responde a criterios de funcionalidad, significatividad, motivacionales y formativos.
2. Las actividades abarcan tanto los diferentes niveles aptitudinales, intereses, género de las personas que participan en ellas, etc.
3. Se estructuran los distintos procesos de trabajo asignando a cada persona responsabilidades dentro de los mismos.
4. Existe una secuenciación de los pasos de las actividades, que aporta una información clara a modo de instrucciones de trabajo comprensibles.

- ***Indicador 51: Las personas con autismo tienen asignadas responsabilidades y participan en la organización.***

Evidencias:

1. Cada persona con autismo tiene asignadas responsabilidades en la organización, adecuadas a sus capacidades e intereses.
2. Existe una revisión periódica que evalúa el grado de adecuación y desarrollo de las responsabilidades asignadas y facilita el reconocimiento hacia las personas con autismo.
3. Existen foros, instrumentos... , para recoger aportaciones y sugerencias de las personas con autismo, fomentando los grupos de autorrepresentación cuando así sea posible.

4. Las personas con autismo participan en el proceso de diseño, creación y adaptación de materiales y actividades.

◦ ***Indicador 52: Se dispone de apoyo y seguimiento técnico integrado en el equipo profesional.***

Evidencias:

1. La organización cuenta con personal técnico especializado en las diferentes áreas de intervención.
2. El trabajo técnico está integrado dentro del equipo, de forma que se planifican y se abordan en grupo las estrategias de intervención individualizadas.
3. Existe un sistema de evaluación continua de los programas específicos de intervención.
4. El seguimiento técnico detecta regularmente nuevas necesidades de intervención o apoyos específicos.

• **Horario:**

◦ ***Indicador 53: El horario y ritmo de trabajo de las personas con autismo se adapta a sus necesidades, intereses y momentos vitales.***

Evidencias:

1. Existen horarios estables y personalizados.
2. Se informa anticipadamente a cada persona con autismo del horario que tiene, así como de los posibles cambios e imprevistos.
3. Se contemplan tiempos en los que cada persona puede desarrollar actividades de libre elección.
4. Existe un ajuste entre tiempo y ritmo de trabajo, y tiempo de descanso.

◦ ***Indicador 54: El horario y distribución de tiempos de los profesionales se adecua a las necesidades de las personas con autismo.***

Evidencias:

1. Existe un horario estable y predecible.
2. El horario y distribución de tiempos garantiza una organización de los tiempos de tránsito entre actividades, de descanso, de entradas y salidas, etc.
3. Ante cambios e imprevistos existe un sistema de reorganización que no afecta a las personas con autismo.

4. Se rentabilizan los recursos personales asignando a los profesionales tareas y funciones, que inciden en la mejora de la calidad del servicio, en los momentos en que no sea necesaria o no tengan asignada atención directa.

- **Comunicación / Coordinación:**

- ***Indicador 55: Se facilita la comunicación entre todas las personas vinculadas a la organización o servicio.***

Evidencias:

1. Existen vías de comunicación formal entre personas vinculadas a la Organización según el ámbito, los implicados y el tema.
2. Todos los profesionales de la organización tienen acceso a los diferentes canales de comunicación.
3. Se utilizan diferentes vías y apoyos para promover la comunicación de las personas con autismo.
4. Existen canales que posibilitan una comunicación continua y rápida entre programas y servicios entre todas las personas implicadas.

- ***Indicador 56: Se contemplan tiempos y espacios para la coordinación.***

Evidencias:

1. Existen tiempos programados para reuniones y coordinación.
2. Existe una organización y planificación que facilita la eficacia de las reuniones: orden del día previo, coordinador, acta, distribución de tiempos por temas, etc.
3. Existe la posibilidad de participar activamente en las reuniones e incorporar temas por parte de todos los participantes.
4. Se realiza un seguimiento de la eficacia de las conclusiones y decisiones que se toman en las reuniones.

- ***Indicador 57: Existe coordinación con otros programas y servicios relacionados con la persona con autismo.***

Evidencias:

1. Existen vías de coordinación entre los diferentes servicios de la organización, en los que participa la persona.

2. Existen vías de coordinación con otros servicios externos a la organización, relacionados con la misma.
3. Existe un proceso que garantiza la gestión y difusión de la información, y el conocimiento a todas las personas, servicios o entidades relacionados con la persona con autismo.
4. Existe un registro de información e incidencias de cada servicio que centralice la información y ayude a coordinarse con el resto de los servicios.

○ ***Indicador 58: Se facilita la comunicación a las personas con autismo.***

Evidencias:

1. Cada persona con autismo tiene definido qué sistema de comunicación utiliza.
2. Todas las personas significativas en la vida de la persona con autismo tienen acceso al conocimiento y uso de los sistemas y/o estrategias de apoyo que ésta utiliza.
3. Se utilizan los sistemas de comunicación con diferentes objetivos: anticipar, facilitar peticiones y deseos, informar, elegir, rechazar, etc.
4. Existe coherencia entre los soportes de estructuración y los de comunicación que se utilizan con cada persona.

• **Evaluación sistemática del servicio y/o la organización:**

○ ***Indicador 59: Se realiza una evaluación interna de la organización.***

Evidencias:

1. La organización se evalúa internamente con instrumentos contrastados.
2. En la evaluación participan las personas con autismo, los profesionales y las familias.
3. Existe una vía para registrar de forma inmediata puntos débiles o situaciones susceptibles de mejora en la organización.
4. Se analizan e implementan acciones de mejora tras la detección de puntos débiles en la organización.

○ ***Indicador 60: La mejora de la organización contempla una evaluación externa.***

Evidencias:

1. En la organización se realizan evaluaciones externas de forma periódica.
2. En la evaluación están implicadas las personas con autismo, profesionales y familias.
3. Los resultados de las evaluaciones se difunden a todas las personas implicadas.
4. De la evaluación externa se derivan mejoras en la organización.

- **Liderazgo:**

- ***Indicador 61: La dirección de la organización impulsa la mejora continua.***

Evidencias:

1. La organización cuenta con un Plan Estratégico que orienta la elaboración de Planes Anuales de Acción o Mejora.
2. La dirección motiva e involucra a las personas con autismo, profesionales y familias en la propuesta y desarrollo de acciones de mejora (grupos de mejora, equipos de transformación, etc.).
3. La dirección está implicada activamente en las acciones de mejora propuestas.
4. La dirección reconoce los esfuerzos y logros de mejora realizados por las personas que la integran.

- **Innovación:**

- ***Indicador 62: La tecnología es un recurso extendido en la organización que favorece mejores apoyos y un mejor desempeño a todos los niveles.***

Evidencias:

1. La organización impulsa o participa en el desarrollo de proyectos tecnológicos que suponen una mejora en la intervención y en la provisión de apoyos.
2. La organización cuenta con recursos tecnológicos y digitales que aumentan su eficiencia y eficacia en la gestión.
3. El programa o servicio utiliza productos de apoyo tecnológico como soporte en la comunicación, intervención y acompañamiento a personas con autismo.
4. La organización desarrolla acciones específicas de formación tecnológica a personas con autismo, familias y profesionales.

- ***Indicador 63: La organización desarrolla procesos de carácter innovador que ayudan a mejorar los apoyos y los servicios, así como a ser más eficaces y eficientes.***

Evidencias:

1. La organización participa en proyectos que suponen transformación y cambio en su forma de hacer, atendiendo a las necesidades que tienen sus grupos de interés y a las oportunidades que descubre en su entorno.
2. La organización y/o el programa o servicio participan en proyectos innovadores con otras organizaciones, países, agentes, etc.
3. La organización cuenta/dispose de un sistema de gestión de la innovación que contempla la planificación, los recursos, desarrollo de productos y evaluación.
4. En los procesos de innovación y creatividad participan y se involucran profesionales con diferentes niveles de responsabilidad.

■ **Recursos y servicios:**

- ***Indicador 64: La organización optimiza los recursos personales.***

Evidencias:

1. El número de profesionales se adecua a las necesidades de ratio de cada persona con autismo y/o actividad.
2. Existen unos criterios de asignación claros de funciones a los profesionales.
3. La organización cuenta con profesionales especializados en las diferentes áreas.
4. Las personas que colaboran de manera voluntaria cuentan con una asignación clara de tiempos y tareas a realizar.

- ***Indicador 65: Existe una adecuada organización del trabajo de los profesionales.***

Evidencias:

1. Existe una estructuración clara de los tiempos, actividades y agrupamientos asignados a cada profesional a lo largo de toda la jornada.

2. Existen tiempos de intervención dedicados a desarrollar programas específicos, personalizados y con profesionales especializados.
3. Existe posibilidad de compatibilizar distintas situaciones de trabajo según las necesidades de las personas con autismo y del momento.
4. Existe una definición y asignación de responsabilidades y funciones dentro de la organización que implica a todos los profesionales.

• ***Indicador 66: Se rentabilizan los recursos materiales.***

Evidencias:

1. Los recursos están a disposición de todos los profesionales.
2. Se incorporan o generan nuevos materiales de apoyo a la intervención, según surgen las necesidades de las personas con autismo.
3. Los recursos materiales y ayudas técnicas permiten la adaptación a las capacidades, intereses y necesidades de apoyo de cada persona con autismo.
4. Los recursos están actualizados, en buen estado, son funcionales, y apropiados a la edad y a las necesidades ergonómicas y de prevención de riesgos de cada persona con autismo.

• ***Indicador 67: El entorno físico favorece la participación, accesibilidad y la autonomía de las personas con autismo.***

Evidencias:

1. Existen suficientes espacios y se adecuan de forma flexible a las necesidades puntuales y cambiantes que pueden tener las personas con autismo.
2. La información/estructuración espacial facilita la comprensión y el desenvolvimiento autónomo y seguro de las personas con autismo.
3. Se minimizan las barreras arquitectónicas, sensoriales o dificultades de acceso a la información.
4. Los espacios se adecuan a la normativa que regula la construcción y favorecen aspectos como la salud y la higiene.

■ **Relación con la comunidad/proyección social:**

- ***Indicador 68: Existen alianzas de colaboración con otras entidades.***

Evidencias:

1. Existen vínculos/convenios con otros recursos y/o entidades relacionadas con los objetivos de la organización o servicio de apoyo.
2. Existen vínculos/convenios con sistemas generales de educación, salud, trabajo y servicios sociales.
3. Existen vínculos/convenios con recursos de la comunidad: centros educativos, empresas, deportes, ocio, etc.
4. Existe un plan de voluntariado (captación, formación y seguimiento).

- ***Indicador 69: La organización asume y comunica un compromiso de responsabilidad social.***

Evidencias:

1. La organización contrata servicios o adquiere productos necesarios, teniendo en cuenta criterios sociales, y medioambientales, entre otros, en los que se favorece el empleo de personas con discapacidad y otras situaciones de vulnerabilidad.
2. La organización garantiza la calidad de los productos o servicios contratados a personas o empresas ajenas, y las condiciones en que estos han sido producidos y realizados.
3. La organización desarrolla acciones positivas relacionadas con su impacto en el medio ambiente.
4. Los espacios se ajustan a los principios del diseño para todas las personas y la accesibilidad universal.

- ***Indicador 70: Se favorece la sensibilización y una imagen social positiva sobre el autismo.***

Evidencias:

1. Se organizan actividades abiertas a toda la comunidad que den respuesta a las necesidades detectadas o demandas recibidas (exposiciones, jornadas, conferencias...).
2. Se participa en proyectos e iniciativas promovidas desde diferentes recursos de la comunidad con el objetivo de favorecer la inclusión y sensibilización social.
3. Se forma a agentes clave y profesionales implicados en la mejora de la calidad de vida de las personas con autismo (salud, educación, operadores jurídicos...).

4. Existe un Plan de Comunicación Externa en el que las personas con autismo tengan protagonismo (material impreso de divulgación, página Web, publicaciones, vídeos, aparición en medios de comunicación, etc.).

3.2. Guía de indicadores reducida

La elaboración de una versión reducida de la Guía de indicadores de calidad de vida para organizaciones y servicios que prestan apoyo a personas con autismo, tiene como objetivo facilitar una administración rápida que permita chequear con mayor agilidad y frecuencia las áreas e indicadores esenciales relacionados con la calidad de la organización o servicio. Se plantea como un instrumento de aproximación, que orienta en el conocimiento de los ámbitos que deben evaluarse con mayor profundidad, no como una herramienta que permite evaluar con precisión. La garantía de calidad de la organización se asegura con la aplicación de la versión extensa del instrumento, y a él remitimos cuando se desee hacer un análisis exhaustivo y riguroso de la misma.

Para evaluar cada indicador, la organización o servicio debe aportar al menos cuatro evidencias por las que considera que este se cumple o está en proceso, e igualmente es imprescindible describir de la manera más concreta posible aquellos aspectos en los que entienden que deben mejorar.

La aplicación de esta versión debe realizarse por parte de un Equipo Evaluador configurado en las mismas condiciones que en la versión extendida, exceptuando la figura del evaluador externo.

La versión consta de 40 indicadores que deben ser evaluados atendiendo a tres categorías: no se cumple, en proceso, y alcanzado. Ello permitirá determinar el perfil de la organización y diseñar el plan de mejora.

Los resultados que se pueden obtener son los siguientes:

Puntuación	Nivel
118-143	Excelente
89-118	Muy bueno
60-88	Bueno
30-59	Mejorable
0-29	Muy mejorable

Figura 3.5: Rangos de puntuación en evaluación simple

Guía de indicadores

- Calidad referida a la persona:
 - Calidad desde la perspectiva de la persona con autismo:
 - Bienestar físico:
 - ◇ *Indicador 1: Existen programas de atención a la salud personalizados y actualizados*
Evidencias:
 - 1.
 - 2.
 - 3.
 - 4.
 - ◇ *Indicador 2: Se garantiza la correcta administración y seguimiento de los tratamientos de salud.*
Evidencias:
 - 1.
 - 2.
 - 3.
 - 4.
 - ◇ *Indicador 3: Se interviene de manera personalizada en el ámbito del cuidado y promoción de la autonomía personal.*
Evidencias:

- 1.
- 2.
- 3.
- 4.

○ Bienestar emocional:

- ◇ *Indicador 4: Se promueve el máximo bienestar emocional en la vida de la persona con autismo.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- ◇ *Indicador 5: Se desarrollan programas personalizados basados en el apoyo conductual positivo.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

○ Bienestar material:

- ◇ *Indicador 6: Se respeta la intimidad y el disfrute de espacios, tiempos y pertenencias personales.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

○ Relaciones interpersonales:

- ◇ *Indicador 7: Se promueven las relaciones sociales significativas y las competencias necesarias para su disfrute.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

○ Desarrollo personal:

- ◇ *Indicador 8: Se promueve el avance y el desarrollo continuo de la persona en diferentes ámbitos de la vida (formación, ocio, laboral, etc.).*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Derechos:**

- ◇ *Indicador 9: Se garantiza el respeto a la identidad y dignidad de la persona.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Autodeterminación:**

- ◇ *Indicador 10: Las personas expresan opiniones, preferencias y toman decisiones significativas sobre sus vidas.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- ◇ *Indicador 11: Las personas con autismo participan en el diseño, implementación y evaluación de sus Planes Individuales de Apoyo.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Inclusión social:**

- ◇ *Indicador 12: Se promueve la inclusión social de las personas con autismo.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- Calidad desde la perspectiva de las familias:

- *Indicador 13: Las actuaciones con la persona con autismo tienen en cuenta a la familia, en los casos que sea pertinente.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- *Indicador 14: Se facilita la implicación y el aumento de la satisfacción de las familias en la organización, en los casos que sea pertinente.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- Calidad desde la perspectiva de los profesionales:

- *Indicador 15: Se conocen, valoran y se tienen en cuenta las propuestas e iniciativas provenientes de los profesionales.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- *Indicador 16: Se facilita la implicación y el aumento de la satisfacción de los profesionales en la organización.*

Evidencias:

- 1.
- 2.

3.

4.

■ **Identificación de las necesidades y preferencias / elaboración y seguimiento de los planes individuales de apoyo:**

• **Planificación:**

- *Indicador 17: Se adecua el proceso para elaborar planes de apoyo con la persona con autismo, adaptados a sus necesidades específicas, capacidades e intereses a lo largo de toda su vida.*

Evidencias:

1.

2.

3.

4.

• **Planificación de apoyos:**

- *Indicador 18: Los apoyos y criterios metodológicos se adaptan a las necesidades y capacidades de la persona con autismo.*

Evidencias:

1.

2.

3.

4.

• **Plan de seguimiento y evaluación:**

- *Indicador 19: Se realiza un seguimiento y evaluación continua de cada Plan Individual de Apoyo.*

Evidencias:

1.

2.

3.

4.

■ **Formación de los profesionales:**

• **Conocimiento del autismo:**

- *Indicador 20: Cada profesional recibe una formación específica sobre autismo y su puesto de trabajo al inicio de su actividad profesional.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- *Indicador 21: Se promueve la formación continua, la actualización y el desarrollo profesional, que incluye aspectos técnicos, organizacionales y valores de la organización.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Conocimiento y adaptación a la persona con autismo:**

- *Indicador 22: Existen mecanismos que garantizan el conocimiento en profundidad de cada persona con autismo.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- *Indicador 23: La intervención de cada profesional se adapta a las necesidades y características de cada persona con autismo.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Actitudes y valores:**

- *Indicador 24: La práctica y actitudes profesionales tienen como referente la misión y los valores de la*

organización o servicio.

Evidencias:

- 1.
- 2.
- 3.
- 4.

- Participación en investigaciones sobre autismo:

- *Indicador 25: La organización promueve la ampliación del conocimiento a través de la participación activa en investigaciones sobre autismo.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- Estructura y organización:

- Grupos de iguales:

- *Indicador 26: La configuración de los grupos de iguales se adapta a las necesidades de las personas con autismo.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- Organización de la actividad:

- *Indicador 27: Las personas con autismo tienen una organización clara y accesible de las tareas y actividades en las que participan.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- *Indicador 28: Las personas con autismo tienen asignadas responsabilidades y participan en la organización.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Horario:**

- *Indicador 29: El horario y ritmo de trabajo de las personas con autismo se adapta a sus necesidades, intereses y momentos vitales.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Comunicación / Coordinación:**

- *Indicador 30: Existen canales que facilitan y promueven la comunicación, así como espacios y tiempos para la coordinación entre todas las personas vinculadas a la organización.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Evaluación sistemática del servicio y/o la organización:**

- *Indicador 31: Se realiza una evaluación interna de la organización.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- *Indicador 32: La mejora de la organización contempla una evaluación externa.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Liderazgo:**

- *Indicador 33: La dirección de la organización impulsa la mejora continua a través de planes específicos en el que participan los diferentes grupos de interés.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Innovación:**

- *Indicador 34: La tecnología es un recurso extendido en la organización que favorece mejores apoyos y un mejor desempeño a todos los niveles.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- *Indicador 35: La organización desarrolla procesos de carácter innovador que ayudan a mejorar los apoyos y los servicios, así como a ser más eficaces y eficientes.*

Evidencias:

- 1.
- 2.
- 3.
- 4.

- **Recursos y servicios:**

- ***Indicador 36: Se optimizan los recursos personales y materiales disponibles dentro de la organización y en el entorno.***

Evidencias:

- 1.
- 2.
- 3.
- 4.

- ***Indicador 37: El entorno físico favorece la participación, accesibilidad y la autonomía de las personas con autismo.***

Evidencias:

- 1.
- 2.
- 3.
- 4.

■ **Relación con la comunidad/proyección social:**

- ***Indicador 38: Existen alianzas de colaboración con otras entidades.***

Evidencias:

- 1.
- 2.
- 3.
- 4.

- ***Indicador 39: La organización o servicio asume y comunica un compromiso de responsabilidad social.***

Evidencias:

- 1.
- 2.
- 3.
- 4.

- ***Indicador 40: Se favorece la sensibilización y una imagen social positiva sobre el autismo.***

Evidencias:

- 1.

- 2.
- 3.
- 4.

Tabulación de datos

[illegible]

Figura 3.6: Tabulación simple de datos

Técnicas y herramientas

4.1. *Microsoft Azure*

Microsoft Azure es una plataforma que proporciona diferentes servicios en la nube, permitiendo la construcción, prueba, despliegue y administración de los mismos. Esta plataforma fue anunciada en el año 2010 por Microsoft con el nombre de *Windows Azure*, pasando a su denominación actual el 25 de marzo del año 2014.

En este proyecto se utiliza *Azure* para desplegar la aplicación web que es utilizada para la gestión de las operaciones de la base de datos. La aplicación web está desplegada mediante el servicio denominado *Web Service*, el cual es utilizado para el despliegue de aplicaciones web hechas en diferentes tecnologías de diferentes lenguajes de programación, destacando C#, Java y Python. Mientras tanto, la base de datos es implementada utilizando *SQL Server* mediante el servicio denominado *Azure SQL*, el cual proporciona la cadena de conexión necesaria para el servicio web, crea el servidor de base de datos y proporciona el soporte necesario para la ejecución de consultas en SQL.

Para poder utilizar *Microsoft Azure* [22], es preciso contar con una cuenta con la cual se tienen ciertos servicios de forma gratuita, algunos de ellos de forma permanente y otros tantos durante un total de 12 meses. Adicionalmente a esta base, se pueden añadir servicios o mejorar los ya existentes a partir de diferentes niveles de suscripción a los mismos, los cuales se ajustan a las necesidades que tengan los usuarios u organizaciones para sus actividades. Para cubrir dichas actividades, *Microsoft Azure* proporciona un crédito inicial de 200\$ para utilizarse durante el primer año, el cual puede ampliarse de forma opcional eligiendo un método de pago, ya sea mediante transferencia bancaria o mediante tarjeta de crédito o de débito.

En primera instancia, el inicio del despliegue de la aplicación en *Microsoft Azure* se ha realizado con la propia cuenta de la Universidad de Burgos [9], el cual permite el uso de *Azure for Education* [20] ya que se trata de uno de los diferentes servicios de *Microsoft 365* del cual disponen los alumnos de manera gratuita, la cual destaca por un crédito inicial de 100\$ y por no necesitar introducir un método de pago para la creación de la cuenta. A posteriori, la implementación definitiva de la aplicación se ha realizado en el propio servidor de la *Fundación Miradas*, disponiendo para ello con una cuenta diferente a la de la Universidad de Burgos.

En cuanto al aprendizaje de la herramienta, se dispone de una herramienta de aprendizaje denominada *Microsoft Learn* [23], la cual consta de diferentes cursos autodidactas e interactivos sin restricción alguna en cuanto a horarios, permitiendo un aprendizaje adaptado al ritmo que cada usuario tenga y al tiempo que éste le pueda dedicar a los mismos. *Microsoft Learn* también dispone de diferentes herramientas alternativas para incrementar la experiencia y el aprendizaje del usuario, como la presencia de foros y la búsqueda de documentación técnica sobre las diferentes herramientas de *Microsoft*. En última instancia, *Microsoft Learn* también ofrece la obtención de diferentes certificados oficiales de *Microsoft*, adaptados al rol que desempeña cada usuario en el equipo de trabajo, todo ello gracias a los cursos y módulos de aprendizaje correspondientes, aunque para conseguir esa certificación se necesita realizar un examen previo pago de una tasa de inscripción al mismo.

4.2. Entornos de desarrollo utilizados

Microsoft Visual Studio

Para el desarrollo del lado del servidor se ha optado por utilizar *Microsoft Visual Studio Community*, la cual se trata de la versión más básica de este entorno de desarrollo, si no consideramos que se tiene *Microsoft Visual Studio Code*. Dicho entorno de desarrollo es gratuito, por lo que no supone ningún coste adicional con respecto a sus versiones *Enterprise* y *Professional*, los cuales sí que tienen una licencia de pago con posibilidad de probar el software de manera gratuita.

Visual Studio es una herramienta de desarrollo eficaz que permite completar todo el ciclo de desarrollo en un solo lugar. Es un entorno de desarrollo integrado completo que permite la escritura, edición, depuración y compilación del código y, luego, su posterior implementación. Aparte de la edición y depuración del código, Visual Studio incluye compiladores, herramientas de

finalización de código, control de código fuente, extensiones y muchas más características para mejorar cada fase del proceso de desarrollo de software. Visual Studio proporciona a los desarrolladores un entorno de desarrollo enriquecido para desarrollar código de alta calidad de forma eficaz y colaborativa:[19]

- Instalador basado en cargas de trabajo: instale solo lo que necesita.
- Herramientas y características de codificación eficaces: todo lo que necesita para compilar sus aplicaciones en un solo lugar.
- Compatibilidad con varios lenguajes: código en C++, C#, JavaScript, TypeScript, Python, etc.
- Desarrollo multiplataforma: compilación de aplicaciones para cualquier plataforma.
- Integración del control de versiones: colaboración en el código con compañeros de equipo.

El motivo por el cual se ha utilizado este entorno de desarrollo para el lado del servidor es por los cursos de *Azure Learn* que se han ido siguiendo para el aprendizaje de las herramientas de Azure para este tipo de aplicaciones, aunque también pueda utilizarse *Microsoft Visual Studio Code* para la programación del lado del servidor. Al haber realizado el aprendizaje de esta manera, no se necesita realizar aprendizaje de otras herramientas, ayudando a reforzar dicha decisión.

Por lo tanto, las principales características de este entorno de desarrollo son las siguientes:[17]

- **Instalación modular:** En el instalador modular de Visual Studio, se eligen y se instalan exclusivamente las cargas de trabajo que sean necesarias. Las cargas de trabajo son grupos de características que los lenguajes de programación o las plataformas necesitan para funcionar. Esta estrategia modular ayuda a reducir la superficie de instalación de Visual Studio, por lo que se instala y actualiza más rápido.
- **Creación de aplicaciones de Azure habilitadas para la nube:** Visual Studio ofrece un conjunto de herramientas para crear fácilmente aplicaciones habilitadas para la nube de Microsoft Azure, permitiendo

la configuración, compilación, depuración, empaquetado e implementación de aplicaciones y servicios de Azure directamente desde el entorno de desarrollo integrado (IDE). Para obtener las plantillas de proyecto y las herramientas de Azure, se tiene que seleccionar la carga de trabajo Desarrollo de Azure al instalar Visual Studio.

- **Creación de aplicaciones web:** Visual Studio puede crear aplicaciones web mediante ASP.NET, Node.js, Python, JavaScript y TypeScript. Visual Studio admite muchos marcos web, como Angular, jQuery y Express. ASP.NET Core y .NET Core funcionan en los sistemas operativos Windows, Mac y Linux. ASP.NET Core es una actualización principal a MVC, WebAPI y SignalR. ASP.NET Core se diseñó desde la base para ofrecer una pila de .NET eficiente y componible, con el fin de compilar servicios y aplicaciones web modernos basados en la nube.
- **Compilar aplicaciones y juegos multiplataforma:** Visual Studio puede crear aplicaciones y juegos para macOS, Linux y Windows, así como para Android, iOS y otros dispositivos móviles. Con Visual Studio, puede crear:
 - Aplicaciones de .NET Core que se ejecutan en Windows, macOS y Linux.
 - Aplicaciones móviles para iOS, Android y Windows en C# y F# mediante Xamarin.
 - Juegos 2D y 3D en C# mediante Visual Studio Tools para Unity.
 - Aplicaciones de C++ nativas para dispositivos iOS, Android y Windows. Comparta código común en bibliotecas para iOS, Android y Windows mediante C++ para desarrollo multiplataforma.
- **Conectarse a bases de datos:** El Explorador de servidores ayuda a explorar y administrar instancias y recursos de servidor de forma local y remota, y en Azure, Microsoft 365, Salesforce.com y sitios web. El Explorador de objetos de SQL Server ofrece una vista de los objetos de base de datos similar a la de SQL Server Management Studio. Con el Explorador de objetos de SQL Server puede realizar trabajos de administración y diseño de bases de datos ligeras. Algunos ejemplos son la edición de datos de tabla, la comparación de esquemas y la ejecución de consultas mediante menús contextuales.
- **Depuración y pruebas:** Con el sistema de depuración de Visual Studio, es posible depurar el código que se ejecuta en el proyecto

local, en un dispositivo remoto o en un emulador de dispositivo. Es posible ejecutar el código una instrucción cada vez, inspeccionando las variables mientras se avanza. O bien, se pueden establecer puntos de interrupción que solo se alcanzan cuando una condición especificada es verdadera. Se pueden administrar las opciones de depuración en el propio editor de código para que no tenga que salir del código. Visual Studio ofrece opciones de prueba, como pruebas unitarias, Live Unit Testing, IntelliTest y pruebas de carga y rendimiento. Visual Studio también cuenta con funciones avanzadas de análisis de código para detectar errores de diseño, de seguridad y de otro tipo.

- **Implementación de la aplicación finalizada:** Visual Studio dispone de herramientas para implementar las aplicaciones en usuarios o clientes mediante Microsoft Store, un sitio de SharePoint o las tecnologías de InstallShield o Windows Installer.
- **Administrar el código fuente:** En Visual Studio, se puede administrar el código fuente en los repositorios de Git hospedados por cualquier proveedor, incluido GitHub. También puede buscar una instancia de Azure DevOps Server a la que conectarse.

Android Studio

Para el desarrollo del lado del cliente se ha decidido utilizar Android Studio, el cual es el entorno de desarrollo integrado oficial de Google para aplicaciones en Android. Desde el 7 de marzo del 2019 Kotlin es el lenguaje de programación preferido de Google para el desarrollo de aplicaciones en Android, aunque esta IDE también permita la implementación de las mismas en el lenguaje Java. [?]

Está basado en el software IntelliJ IDEA de JetBrains y ha sido publicado de forma gratuita a través de la Licencia Apache 2.0. Está disponible para las plataformas GNU/Linux, macOS, Microsoft Windows y Chrome OS. Ha sido diseñado específicamente para el desarrollo de Android.

Como lenguaje de programación se ha utilizado Java, ya que es un lenguaje que se ha utilizado a lo largo de la carrera en diferentes asignaturas, siendo uno de los lenguajes de programación más utilizados en los últimos años. Para el desarrollo del lado del cliente se ha optado por este entorno de desarrollo debido a que ya se sabía manejar de la asignatura de *Interacción Hombre-Máquina* del cuarto semestre de este grado, aun sabiendo que se tenía la opción de utilizar el mismo entorno que en el lado del servidor.

Las características de la versión más reciente de Android Studio a fecha de la entrega de segunda convocatoria, teniendo en cuenta que siempre se añaden nuevas funcionalidades en cada una de sus versiones, son las siguientes:[24]

- El soporte para la construcción de las aplicaciones está basado en Gradle, el cual ayuda a automatizar y administrar el proceso de compilación de las mismas mediante las dependencias que va añadiendo el usuario.
- La refactorización del código y de su estructura es específica de Android, teniendo también la posibilidad de realizar arreglos rápidos.
- Posee también herramientas Lint para detectar problemas de rendimiento, usabilidad, compatibilidad de versiones y otros problemas. Dichas herramientas han sido de gran utilidad para poder detectar los diferentes errores que han impedido que la aplicación se mostrase de la manera adecuada
- Integración de ProGuard y funciones de firma de aplicaciones. ProGuard es utilizado para la reducción y optimización del código de la aplicación del código, con la finalidad de que el rendimiento sea óptimo en los dispositivos móviles en los que se ejecuta la aplicación.
- Android Studio cuenta también con diferentes plantillas para crear diseños comunes de Android y otros componentes, pudiendo modificarse mediante un editor de diseño enriquecido que permite a los usuarios arrastrar y soltar componentes de la interfaz de usuario. Esta característica es fundamental para ayudar al desarrollador a elegir los mejores diseños base para las diferentes actividades de su aplicación, por lo que no es necesario tener amplios conocimientos en el lenguaje XML para empezar a desarrollarla.
- Android Studio también tiene soporte para programar aplicaciones para diferentes dispositivos, entre los cuales destacamos los teléfonos móviles, las tabletas, las aplicaciones de escritorio y los dispositivos de Android Wear.
- Android Studio tiene soporte integrado para Google Cloud Platform, que permite la integración con Firebase Cloud Messaging (antes 'Google Cloud Messaging') y Google App Engine.

- Para realizar las pruebas de la aplicación se cuenta con un dispositivo virtual de Android, teniendo también el soporte para la depuración inalámbrica para dispositivos físicos.
- El renderizado se realiza en tiempo real.
- Android Studio tiene su propia consola de desarrollador, además de tener la capacidad de integrar diferentes terminales dependiendo del sistema operativo que se esté utilizando.

Apoyándonos en las características anteriormente mencionadas, las principales ventajas de utilizar Android Studio son las siguientes:

- Como se ha mencionado con anterioridad, es la IDE oficial de Google para el desarrollo de aplicaciones de Android, desbancando a Eclipse en el año 2013.
- Permite la conversión de código Java a código Kotlin, algo que es imposible en otras IDEs como Eclipse, ya que Kotlin es un lenguaje el cual se ejecuta sobre una máquina virtual de Java, permitiendo también utilizar sus librerías.
- Permite programar la interfaz de la aplicación de forma interactiva, todo ello mediante los archivos .xml del directorio /res/layout, pudiendo intercalar de forma sencilla entre la forma interactiva y el código .xml.
- Permite simular el funcionamiento de la aplicación sobre diferentes dispositivos, ya sean virtuales mediante su emulador, o físicos pudiendo conectar diferentes dispositivos mediante las opciones de desarrollador de los dispositivos Android.
- Permite inicializar proyectos a partir de plantillas preestablecidas, siendo de gran utilidad tanto para principiantes como para expertos.
- Permite la creación de módulos de Java, no sólo de módulos de Android, permitiendo así ejecutar esos módulos a parte para el desarrollo de diferentes pruebas para la versión básica de la ejecución del código de indicadores en línea de comandos.

En contraparte, los principales inconvenientes de Android Studio son los siguientes:

- Android Studio dificulta mucho la unificación del desarrollo de aplicaciones cliente-servidor bajo un mismo entorno, ya que resulta muy tedioso ejecutar tanto la aplicación como el servidor en dos hilos diferentes.
- Android Studio no soporta otros lenguajes de programación diferentes de Java y Kotlin, por lo tanto no se puede programar el servidor de *ASP.NET* en C#, obligando al uso de otra IDE distinta para su implementación, como *Visual Studio 2022*.
- El emulador de Android Studio en ocasiones tiene un desempeño que deja mucho que desear debido a su inestabilidad en tiempo de ejecución, sucediendo lo mismo con el desarrollo de los layout de las actividades, que en los modos *Split* y *Design* tiene diferentes problemas de renderización. Afortunadamente estos problemas no se han trasladado a los dispositivos físicos en los que se han realizado las pruebas, solventando el pobre desempeño que pueda tener el emulador.
- Android Studio no tiene un soporte nativo de Azure debido a que Android Studio es de Google y Azure es de Microsoft. Eso obliga al usuario a tener que buscar otro entorno diferenciado para poder realizar la implementación del lado del servidor, algo que se tuvo que hacer casi al final del tiempo de desarrollo del mismo, cuando se pasó de tratar de implementar un servidor embebido en la aplicación para la realización de pruebas del servidor en local, obligando a hacer este cambio para avanzar con el desarrollo

4.3. Lenguajes de programación y herramientas utilizadas

Lado del cliente

Java

En el lado del cliente se ha utilizado Java como lenguaje de programación debido a que, como se ha mencionado en la sección anterior, dicho lenguaje ya se había utilizado en las asignaturas del grado de Metodología de la Programación, Estructuras de Datos, Interacción Hombre-Máquina, Programación Concurrente y de Tiempo Real, Aplicaciones de Bases de Datos, Testes e Qualidade de Software (equivalente en el *Instituto Superior de Engenharia de Coimbra* a la asignatura Validación de Datos de esta

universidad) y Sistemas Distribuidos, además de las asignaturas de máster Arquitectura y servicios de internet y Sistemas de Información Avanzados. Por lo que esta decisión viene respaldada por la experiencia otorgada por los docentes de dichas asignaturas durante todo el grado.

Java es un lenguaje de programación y una plataforma informática que fue comercializada por primera vez en 1995 por Sun Microsystems.

El lenguaje de programación Java fue desarrollado originalmente por James Gosling, de Sun Microsystems (en la actualidad propiedad de Oracle), y publicado en 1995 como un componente fundamental de la plataforma Java de Sun Microsystems. Su sintaxis deriva en gran medida de C y C++, pero tiene menos utilidades de bajo nivel que cualquiera de ellos. Las aplicaciones de Java son compiladas a bytecode (clase Java), que puede ejecutarse en cualquier máquina virtual Java (JVM) sin importar la arquitectura de la computadora subyacente.

Por lo tanto, para el lado del cliente se han utilizado las siguientes herramientas de Java:

- `java.util.concurrent`
- *OKHttp*: OkHttp es un cliente HTTP que es eficiente por defecto, ya que:
 - El soporte de HTTP/2 permite que todas las solicitudes al mismo servidor compartan un socket.
 - La agrupación de conexiones reduce la latencia de las solicitudes (si no está disponible HTTP/2).
 - La compresión transparente GZIP reduce el tamaño de las descargas.
 - La caché de respuestas evita completamente la red en las solicitudes repetidas.
 - OkHttp persevera cuando la red tiene problemas: se recuperará silenciosamente de problemas de conexión comunes. Si tu servicio tiene múltiples direcciones IP, OkHttp intentará con direcciones alternativas si la primera conexión falla. Esto es necesario para IPv4+IPv6 y servicios alojados en centros de datos redundantes. OkHttp admite funciones TLS modernas (TLS 1.3, ALPN, verificación de certificado). Se puede configurar para que tenga una conexión alternativa para una amplia conectividad.

Usar *OkHttp* es fácil. Su API de solicitud/respuesta está diseñada con constructores fluidos e inmutabilidad. Admite tanto llamadas de bloqueo síncronas como llamadas asíncronas con devoluciones de llamada.

En este caso *OkHttp* se utiliza en conjunto con *Retrofit* para la construcción de un cliente nuevo mediante la función `OkHttpClient().newBuilder().build()`:

```
1      client=new OkHttpClient().newBuilder().build();
```

- *Retrofit*: *Retrofit* es la clase a través de la cual las interfaces de API se convierten en objetos invocables. Por defecto, *Retrofit* proporciona valores predeterminados sensatos, pero también permite personalización. Por defecto, *Retrofit* solo puede deserializar cuerpos HTTP en el tipo `ResponseBody` de *OkHttp* y solo puede aceptar su tipo `RequestBody` para la anotación `@Body`. Por lo tanto, un ejemplo de *Retrofit* es el siguiente:

```
1      Retrofit retrofit = new Retrofit.Builder()
2          .baseUrl("https://api.github.com/")
3          .addConverterFactory(GsonConverterFactory.
4              ↪ create())
5          .build();
```

Como se ha mencionado con anterioridad, *Retrofit* se utiliza en conjunto con *OkHttp* para poder enviar las peticiones al servidor y poder recibir posteriormente sus respuestas. *Retrofit* proporciona las etiquetas necesarias para indicar a las APIs el tipo de consulta a realizar `@GET`, `@POST`, `@PUT` y `@DELETE`, el cuerpo a enviar junto con la solicitud `@Body` y los atributos a añadir al path `@Path`.

Lado del servidor

C#

Para el lado del servidor se ha decidido utilizar C# como lenguaje de programación, aunque en la gran mayoría de la fase de desarrollo se haya pretendido utilizar el mismo lenguaje para implementar toda la aplicación cliente servidor, utilizando JDBC para la conexión de la base de datos y JAX-RS como API para la aplicación web, se ha optado finalmente por C# junto con el framework de ASP.NET debido a que dicho lenguaje y dicho framework tienen el soporte integrado en *Azure* para la implementación y posterior despliegue de la aplicación web que soporta la base de datos, lo que ha hecho que los tiempos de respuesta de las solicitudes sean bastante más cortos en comparación con la alternativa anteriormente mencionada basada en Java.

C# es un lenguaje de programación multiparadigma desarrollado y estandarizado por la empresa Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270). C# es uno de los lenguajes de programación diseñados para la infraestructura de lenguaje común. Su sintaxis básica está basada en C y C++, utilizando también el modelo de objetos de la plataforma .NET, similar al de Java, lo que ha favorecido al rápido aprendizaje de este lenguaje.[25]

[18] *ASP.NET Core* es un marco multiplataforma de código abierto y de alto rendimiento cuyo fin es compilar aplicaciones que se encuentran en internet o en la nube. Este marco da la posibilidad de :

- Compilar servicios y aplicaciones web, aplicaciones de Internet de las cosas (IoT) y back-ends móviles.
- Efectuar implementaciones locales y en la nube.
- Ejecutar en .NET Core.

Las principales ventajas que tiene *ASP.NET* con respecto a su competencia son las siguientes:

- Da la posibilidad de crear una aplicación cliente-servidor de forma unificada, aunque este no haya sido el caso.
- Está diseñado para realizar pruebas
- ASP.NET dispone de las Razor Pages, que se trata un modelo de programación basado en páginas que facilita la compilación de interfaces de usuario web y hace que sea más productiva.
- Blazor permite usar C# en el explorador, junto con JavaScript, permitiendo compartir la lógica entre el cliente y el servidor.
- Capacidad para desarrollarse y ejecutarse en cualquier sistema operativo.
- De código abierto y centrado en la comunidad.
- Integración de marcos del lado cliente modernos y flujos de trabajo de desarrollo.

- Compatibilidad con el hospedaje de servicios de llamada a procedimiento remoto con gRPC.
- Un sistema de configuración basado en el entorno y preparado para la nube.
- Tiene la inserción de dependencias integrada.
- Tiene una canalización de solicitudes HTTP ligera, modular y de alto rendimiento.
- Tiene la capacidad de hospedar diferentes tipos de servidores.
- Control de versiones en paralelo.
- Herramientas que simplifican el desarrollo web moderno.

Inicio de sesión

Para poder iniciar sesión, se ha creado un método POST denominado **login**: El método **Login** es un controlador en ASP.NET que maneja peticiones HTTP POST en la ruta "login". Este método permite a los usuarios autenticarse proporcionando sus credenciales en formato JSON. A continuación se describe su funcionamiento, destacando especialmente cómo se genera el token y bajo qué condiciones se generan.

- El método está decorado con los atributos `[HttpPost("login")]` y `[AllowAnonymous]`, lo que indica que este método responderá a las solicitudes POST en la ruta "login" y no requiere autenticación previa para acceder.
- El método recibe un parámetro **credentials** del cuerpo de la solicitud (`[FromBody]`), que es un documento JSON. Se espera que este JSON contenga las propiedades **email** y **password**.

- Dentro del bloque **try**, se extraen el **email** y **password** del **credentials**:

```
1      string email = credentials.RootElement.GetProperty("email "
    ↪      ).ToString();
2      string password = credentials.RootElement.GetProperty("
    ↪      password").ToString();
```

- Se busca al usuario en la base de datos utilizando el contexto **_context**:

```
1      var user = _context.Users.FirstOrDefault(u => u.emailUser
    ↪      == email && u.passwordUser == password && u.isActive
    ↪      == 1);
```

- Si el usuario no se encuentra o no está activo (`isActive == 1`), se realiza otra búsqueda solo por `email` para verificar si el usuario existe:
 - Si no se encuentra ningún usuario con ese `email`, se devuelve un `NotFound()`.
 - Si el `password` es incorrecto, se devuelve un `Unauthorized()`.
- Si el usuario es encontrado y las credenciales son correctas, se procede a generar un token JWT:

```

1      var tokenHandler = new JwtSecurityTokenHandler();
2      var key = Encoding.ASCII.GetBytes(_sessionConfig.secret);
3      var tokenDescriptor = new SecurityTokenDescriptor
4      {
5          Subject = new ClaimsIdentity(new Claim[]
6          {
7              new Claim(ClaimTypes.Name, user.emailUser),
8              new Claim(ClaimTypes.Role, user.userType)
9          }),
10         Expires = DateTime.UtcNow.AddHours(12),
11         SigningCredentials = new SigningCredentials(new
            ↳ SymmetricSecurityKey(key), SecurityAlgorithms.
            ↳ HmacSha256Signature)
12     };
13     var token = tokenHandler.CreateToken(tokenDescriptor);
14     string sessionToken = tokenHandler.WriteToken(token);
    
```

- El token JWT se genera usando el `JwtSecurityTokenHandler` y contiene los siguientes elementos:
 - `ClaimsIdentity` con dos claims: el nombre del usuario (`ClaimTypes.Name`) y el rol del usuario (`ClaimTypes.Role`).
 - La fecha de expiración del token, configurada para 12 horas desde la generación.
 - Credenciales de firma (`SigningCredentials`) utilizando una clave simétrica (`SymmetricSecurityKey`) y el algoritmo HMAC-SHA256.
- Se crea un objeto `usr` que contiene información del usuario autenticado.
- Se busca la organización asociada al usuario:

```

1      var organization = _context.Organizations.FirstOrDefault(o
            ↳ => o.idOrganization == user.idOrganization && o.
            ↳ orgType == user.orgType && o.illness == user.illness
            ↳ );
    
```

- Si no se encuentra la organización, se devuelve un `BadRequest()`.

- Se crea un objeto **org** con la información de la organización.
- Finalmente, se construye una respuesta que contiene la información del usuario, la información de la organización y el token JWT:

```

1      var response = new
2      {
3          user = usr,
4          organization = org,
5          token = "Bearer " + sessionToken
6      };
7      return Ok(response);

```

En resumen, el método **Login** autentica al usuario mediante sus credenciales, genera un token JWT si las credenciales son correctas y el usuario está activo, y devuelve una respuesta que incluye la información del usuario, la organización y el token JWT.

El proceso de autorización de acceso a ciertos métodos en una aplicación ASP.NET, después de que un usuario obtiene un token JWT en el login, se maneja mediante políticas de autorización definidas en el archivo **Program.cs** y los atributos de autorización en los controladores. Aquí está una explicación detallada del funcionamiento:

- **Definición de Políticas de Autorización:** En **Program.cs**, se definen varias políticas de autorización con roles específicos:

```

1      builder.Services.AddAuthorization(options =>
2      {
3          options.AddPolicy("Administrator", policy =>
4              policy.RequireRole("ADMIN"));
5
6          options.AddPolicy("Director", policy =>
7              policy.RequireRole("DIRECTOR"));
8
9          options.AddPolicy("Organization", policy =>
10             policy.RequireRole("ORGANIZATION"));
11     });

```

- Política **Administrator**: Solo los usuarios con el rol ADMIN pueden acceder a métodos que requieren esta política.
 - Política **Director**: Solo los usuarios con el rol DIRECTOR pueden acceder a métodos que requieren esta política.
 - Política **Organization**: Solo los usuarios con el rol ORGANIZATION pueden acceder a métodos que requieren esta política.
- **Uso de los Métodos con Autorización:**

- Este método está marcado con el atributo `[AllowAnonymous]`, lo que significa que no requiere autenticación ni autorización:

```

1      [HttpGet("get")]
2      [AllowAnonymous]
3      public ActionResult<Organization> Get([
        ↳ FromQuery] int id, [FromQuery] string
        ↳ orgType, [FromQuery] string illness)
4      {
5          // Implementacion del metodo
6      }
```

- Este método está protegido por la política `Administrator`:

```

1      [HttpPost]
2      [Authorize(Policy = "Administrator")]
3      public IActionResult Create([FromBody]
        ↳ Organization organization, [FromHeader]
        ↳ string Authorization)
4      {
5          // Implementacion del metodo
6      }
```

Solo los usuarios con el rol `ADMIN` pueden acceder a este método. El cliente debe enviar el token `JWT` en el encabezado de la solicitud `Authorization`, y el servidor verificará si el token es válido y contiene el rol necesario.

- Este método está protegido por la política `Director`:

```

1      [HttpPut]
2      [Authorize(Policy = "Director")]
3      public IActionResult Update([FromQuery] int
        ↳ id, [FromQuery] string orgType, [
        ↳ FromQuery] string illness, [FromBody]
        ↳ Organization organization, [FromHeader]
        ↳ string Authorization)
4      {
5          // Implementacion del metodo
6      }
```

Solo los usuarios con el rol `DIRECTOR` pueden acceder a este método. El cliente debe enviar el token `JWT` en el encabezado de la solicitud `Authorization`, y el servidor verificará si el token es válido y contiene el rol necesario.

Por tanto, el flujo de ejecución es el siguiente:

1. Login y Obtención del Token:

- El cliente envía una solicitud de login con las credenciales del usuario.

- Si las credenciales son correctas, el servidor genera un token JWT que incluye información sobre el usuario y sus roles.
- El token se devuelve al cliente.

2. Acceso a Métodos Protegidos:

- Para acceder a métodos protegidos (Create, Update), el cliente debe incluir el token JWT en el encabezado Authorization de la solicitud, en formato Bearer <token>.
- El servidor extrae el token del encabezado y lo valida:
 - Verifica la firma del token.
 - Verifica la fecha de expiración del token.
 - Verifica que el token contenga los roles necesarios según la política aplicada al método.
- Si el token es válido y contiene los roles necesarios, el servidor permite el acceso al método.
- Si el token es inválido o no contiene los roles necesarios, el servidor devuelve un error de autorización (por ejemplo, 401 Unauthorized o 403 Forbidden).

Azure SQL

[21] *Azure SQL Database* es un motor de base de datos de plataforma como servicio (PaaS) totalmente administrado que se encarga de la mayoría de las funciones de administración de bases de datos, incluidas *la supervisión sin intervención del usuario, la aplicación de revisiones, la creación de copias de seguridad y la actualización*. El motor de base de datos de Azure SQL Server se ejecuta siempre en la versión más reciente y estable del motor de base de datos de SQL Server, así como en un sistema operativo revisado que tiene una disponibilidad del 99,99 %. Las funcionalidades de PaaS en Azure SQL Database permiten concentrarse en las actividades de administración y optimización de bases de datos específicas del dominio que son importantes para el negocio.

El motor de base de datos más reciente de *Microsoft SQL Server* es la base de *Azure SQL Database*, permitiendo utilizar características avanzadas de procesamiento de consultas, como el procesamiento de consultas inteligente y las tecnologías de memoria de alto rendimiento. De hecho, las últimas funcionalidades de SQL Server se publican primero en Azure SQL Database

y luego en SQL Server mismo. Las funcionalidades más recientes de SQL Server se pueden obtener sin costo mediante actualizaciones o revisiones, y se han probado en millones de bases de datos.

En cuanto a las opciones de implementación de base de datos de Azure SQL, podemos resaltar dos:

- **Una base de datos única** es una base de datos aislada que se administra completamente. Si tiene aplicaciones y microservicios modernos en la nube que necesitan un solo origen de datos confiable, puede usar esta opción. Una sola base de datos es similar al motor de base de datos de SQL Server. Es la opción más sencilla para nuestro caso, sobre todo al principio del desarrollo.
- El **grupo elástico** es una colección de bases de datos distintas con un conjunto compartido de recursos, como la memoria o la CPU. Un grupo elástico puede permitir el movimiento de una sola base de datos. Éste no se ha implementado en esta aplicación, aunque

4.4. Sistema experto

[15] Los sistemas expertos, enmarcados en el campo de la Inteligencia Artificial (IA), son programas informáticos que emulan la capacidad de una persona experta en un dominio de conocimiento o área de actividad específica.

Su evolución y historia se remontan a las décadas de 1960 y 1970, cuando los pioneros en la IA, como Allen Newell y Herbert A. Simon, desarrollaron los primeros sistemas de razonamiento basados en reglas. Estos sistemas se centraron en la resolución de problemas mediante la aplicación de lógica inductiva y deductiva.

A medida que avanzaba el tiempo, los sistemas de conocimiento experimentaron un crecimiento significativo en la década de 1980, con el desarrollo de herramientas como MYCIN para el diagnóstico médico y Dendral para la identificación de compuestos químicos. Esta era también vio la creación de herramientas de desarrollo de sistemas expertos, como CLIPS. Sin embargo, a medida que avanzaba la década de 1990, los sistemas expertos enfrentaron desafíos debido a su dependencia en la representación de conocimiento y la captura de reglas, lo que llevó a una mayor exploración de otras ramas de la IA, como el aprendizaje automático. A pesar de esto, los sistemas expertos siguen siendo una parte importante de la IA, y con la evolución tecnológica actual, continúan desempeñando un papel relevante en la toma de decisiones

automatizada.

La tecnología subyacente que impulsa a los sistemas expertos se basa en varios conceptos fundamentales, entre los que se incluyen:

- **Reglas Heurísticas:** Las reglas heurísticas son pautas o principios empíricos que guían el razonamiento humano. En los sistemas expertos, se utilizan reglas heurísticas para codificar el conocimiento de expertos en forma de *si-entonces*. Por ejemplo, *Si se han marcado 4 evidencias en el indicador, el indicador se marca como conseguido*. Estas reglas permiten que el sistema experto tome decisiones basadas en la experiencia acumulada.
- **Lógica Difusa:** La lógica difusa es un enfoque que permite manejar la incertidumbre y la imprecisión en el razonamiento. A diferencia de la lógica binaria convencional (verdadero o falso), la lógica difusa permite representar grados de verdad. Esto es especialmente útil cuando se trata con conceptos vagos o subjetivos. Por ejemplo, en la obtención del nivel de la evaluación de indicadores, obtenemos cinco niveles a partir de la puntuación total, los cuales son: *excelente*, *muy bueno*, *bueno*, *mejorable* y *muy mejorable*, lo cual ayuda a determinar cómo de bien se ha cumplido la evaluación de indicadores, más allá de determinar si se ha cumplido o no la misma evaluación de indicadores.
- **Redes Bayesianas:** Las redes bayesianas son modelos probabilísticos que utilizan teoremas de probabilidad bayesiana para representar y actualizar la incertidumbre en un sistema. Son especialmente útiles cuando se deben tomar decisiones basadas en evidencia acumulada. Por ejemplo, en medicina, una red bayesiana puede ayudar a calcular la probabilidad de que un paciente tenga una enfermedad en función de múltiples factores, como síntomas, historial médico y resultados de pruebas.

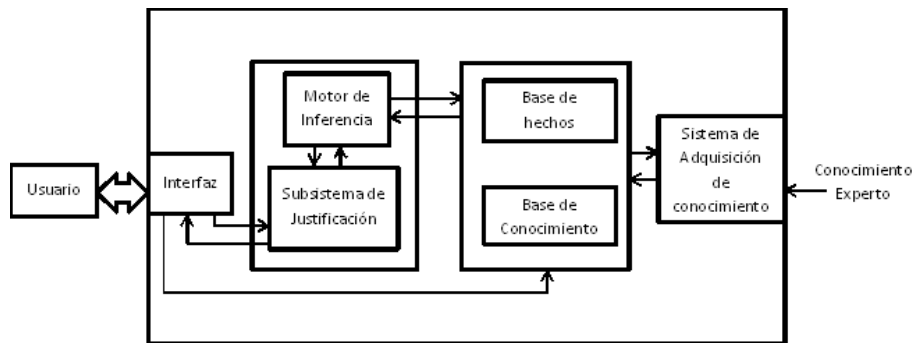


Figura 4.1: Estructura de un sistema experto

La estructura básica de un sistema experto tiene los siguientes componentes:

- **Usuario:** Es la persona o el sistema que interactúa con el sistema experto para obtener soluciones a problemas específicos. En nuestro caso se trata del administrador de *Fundación Miradas*, quien se encarga de proporcionar los datos necesarios sobre las evidencias para que el sistema experto comience a trabajar.
- **Interfaz:** Es el medio por el cual el usuario interactúa con el sistema experto. En nuestro caso es la interfaz gráfica de la aplicación Android, lugar donde el administrador de *Fundación Miradas* se encarga de señalar las evidencias que se cumplen de cada uno de los indicadores.
- **Motor de Inferencia:** Es el corazón del sistema experto. Utiliza las reglas y datos en la base de conocimientos para inferir conclusiones. También interactúa con la base de hechos.
- **Base de hechos:** Contiene datos o hechos específicos del problema actual que el usuario está tratando de resolver. En nuestro caso son por los registros de los indicadores y la evaluación de indicadores correspondiente, los cuales contienen toda la información necesaria para que sea proporcionada al motor de inferencia.
- **Base de Conocimiento:** Contiene el conocimiento experto en forma de hechos y reglas. Este conocimiento es generalmente adquirido de expertos humanos y codificado en una forma que el motor de inferencia puede utilizar. En nuestro caso contiene todo el conjunto de reglas sobre estado de los indicadores a partir del número de evidencias marcadas, si dichos indicadores requieren estar en el plan de mejora,

la puntuación para cada combinación interés del indicador-estado del indicador y el nivel de puntuación de la evaluación de indicadores.

- **Sistema de Adquisición de Conocimiento:** Es el componente que facilita la adquisición de conocimiento de los expertos humanos y su incorporación a la base de conocimientos.
- **Subsistema de Justificación:** Proporciona explicaciones sobre cómo se llegó a una determinada conclusión o por qué se pide cierta información. Esto aumenta la confianza del usuario en las conclusiones del sistema.
- **Conocimiento Experto:** Es el conocimiento que los expertos humanos tienen sobre el dominio del problema. Este conocimiento es adquirido por el sistema de adquisición de conocimiento y almacenado en la base de conocimientos.

En nuestro caso, utilizaremos un **sistema basado en reglas (SBR)**, que son sistemas basados en reglas clásicas bien conocidas por el mundo de la informática en la forma IF (condición) y THEN (acción). Dado un conjunto de hechos, los sistemas de conocimiento son capaces de deducir nuevos hechos gracias a sus reglas.

Por ejemplo, para nuestro caso con la evaluación de indicadores, se ha aplicado para el cálculo de resultados y para determinar el nivel de los mismos:

1. Por un lado, se tiene que determinar qué indicadores necesitan estar incluidos en el plan de mejora y qué indicadores no necesitan estar en el plan de mejora, aplicando las reglas en dos fases:
 - a) En primer lugar, se comprueba para cada indicador el número de evidencias marcadas, teniendo las siguientes reglas para ello:
 - IF indicadorN.numEvidenciasCumplidas=0
OR
indicadorN.numEvidenciasCumplidas=1
THEN
indicadorN.estado=en_comienzo
 - IF indicadorN.numEvidenciasCumplidas=2
OR
indicadorN.numEvidenciasCumplidas=3
THEN
indicadorN.estado=en_proceso

- IF indicadorN.numEvidenciasCumplidas=4
THEN
indicadorN.estado=conseguido

b) Posteriormente, a partir del estado, determinamos si un indicador requiere o no requiere estar en el plan de mejora:

- IF indicadorN.estado=en_comienzo
OR
indicadorN.estado=en_proceso
THEN
indicadorN.necesitaEstarEnPlanDeMejora=SI
- IF indicadorN.estado=conseguido
THEN
indicadorN.necesitaEstarEnPlanDeMejora=NO

2. Por otro lado, a partir de los resultados se determina la puntuación para cada combinación de interés y de estado de indicador, y a partir de ahí obtener el nivel:

a) En primer lugar obtenemos el valor del indicador para añadirlo a la suma total y de paso para calcular la puntuación para cada combinación de interés y de estado de indicador:

- IF indicadorN.estado=conseguido
AND
indicadorN.interes=interes_fundamental
THEN
indicadorN.valor=5
- IF indicadorN.estado=en_proceso
AND
indicadorN.interes=interes_fundamental
THEN
indicadorN.valor=4
- IF indicadorN.estado=conseguido
AND
indicadorN.interes=interes_alto
THEN
indicadorN.valor=4
- IF indicadorN.estado=en_proceso
AND
indicadorN.interes=interes_alto

```

THEN
    indicadorN.valor=3
■ IF indicadorN.estado=conseguido
  AND
    indicadorN.interes=interes_medio
  THEN
    indicadorN.valor=3
■ IF indicadorN.estado=en_proceso
  AND
    indicadorN.interes=interes_medio
  THEN
    indicadorN.valor=2
■ IF indicadorN.estado=conseguido
  AND
    indicadorN.interes=interes_bajo
  THEN
    indicadorN.valor=2
■ IF indicadorN.estado=en_proceso
  AND
    indicadorN.interes=interes_bajo
  THEN
    indicadorN.valor=1

```

b) Posteriormente, a partir del tipo de evaluación y de la puntuación total, se obtiene un nivel del test de indicadores:

```

■ IF testIndicadores.tipo=completo
  AND
    testIndicadores.puntuacion>=200
  OR
    testIndicadores.puntuacion<=250
  THEN
    testIndicadores.nivel=excelente
■ IF testIndicadores.tipo=completo
  AND
    testIndicadores.puntuacion>=150
  OR
    testIndicadores.puntuacion<=200
  THEN
    testIndicadores.nivel=muy_bueno
■ IF testIndicadores.tipo=completo
  AND

```

```
testIndicadores.puntuacion>=100
OR
testIndicadores.puntuacion<=150
THEN
testIndicadores.nivel=bueno
■ IF testIndicadores.tipo=completo
AND
testIndicadores.puntuacion>=50
OR
testIndicadores.puntuacion<=100
THEN
testIndicadores.nivel=mejorable
■ IF testIndicadores.tipo=completo
AND
testIndicadores.puntuacion>=0
OR
testIndicadores.puntuacion<=50
THEN
testIndicadores.nivel=muy_mejorable
■ IF testIndicadores.tipo=simple
AND
testIndicadores.puntuacion>=118
OR
testIndicadores.puntuacion<=143
THEN
testIndicadores.nivel=excelente
■ IF testIndicadores.tipo=simple
AND
testIndicadores.puntuacion>=89
OR
testIndicadores.puntuacion<=118
THEN
testIndicadores.nivel=muy_bueno
■ IF testIndicadores.tipo=simple
AND
testIndicadores.puntuacion>=60
OR
testIndicadores.puntuacion<=88
THEN
testIndicadores.nivel=bueno
```

```

■ IF testIndicadores.tipo=simple
  AND
  testIndicadores.puntuacion>=30
  OR
  testIndicadores.puntuacion<=59
  THEN
  testIndicadores.nivel=mejorable
■ IF testIndicadores.tipo=simple
  AND
  testIndicadores.puntuacion>=0
  OR
  testIndicadores.puntuacion<=29
  THEN
  testIndicadores.nivel=muy_mejorable

```

Para poder programar el sistema experto, se utilizó **NRules** en el lado del servidor.

NRules

NRules es un motor de reglas de código abierto para .NET que se basa en el algoritmo de coincidencia Rete. Las reglas se crean en C# mediante DSL interno.^[7]

NRules también es un motor de inferencia donde, a diferencia de los motores de secuencias de comandos, no existe un orden predefinido en el que se ejecutan las reglas. En cambio, el motor de inferencia determina qué reglas deben activarse en función de los hechos que se le proporcionan y luego las ejecuta de acuerdo con un algoritmo de resolución de conflictos. Entre otras características, NRules admite encadenamiento directo, consultas de hechos complejos y cuantificadores negativos, existenciales y universales.^[7]

Antes de nada, debemos añadir el paquete de nuget de la siguiente manera:^[7]

```
1 dotnet add package NRules
```

El uso básico de NRules se resume en diferentes pasos:^[1]

- En primer lugar, debemos crear diferentes modelos, algo que ya tenemos hecho. NRules sugiere en su documentación como comienzo crear una clase **Customer** y una clase llamada **Order**:^[1]

```

1 public class Customer
2 {
3     public string Name { get; }

```

```

4         public bool IsPreferred { get; set; }
5
6         public Customer(string name)
7         {
8             Name = name;
9         }
10
11        public void NotifyAboutDiscount()
12        {
13            Console.WriteLine($"Customer {Name} was
14                               ↪ notified about a discount");
15        }
16
17        public class Order
18        {
19            public int Id { get; }
20            public Customer Customer { get; }
21            public int Quantity { get; }
22            public double UnitPrice { get; }
23            public double PercentDiscount { get; set; }
24            public bool IsOpen { get; set; } = true;
25
26            public Order(int id, Customer customer, int
27                               ↪ quantity, double unitPrice)
28            {
29                Id = id;
30                Customer = customer;
31                Quantity = quantity;
32                UnitPrice = unitPrice;
33            }

```

- Posteriormente, con estos modelos, se crean diferentes reglas. Por ejemplo, una regla que determina que si en caso de que un cliente es preferente se le asigna un descuento del 10 %:[1]

```

1         public class PreferredCustomerDiscountRule : Rule
2         {
3             public override void Define()
4             {
5                 Customer customer = default;
6                 IEnumerable<Order> orders = default;
7
8                 When()
9                     .Match<Customer>(() => customer, c => c.
10                               ↪ IsPreferred)
11                     .Query(() => orders, x => x
12                       .Match<Order>(
13                         o => o.Customer == customer,
14                         o => o.IsOpen,
15                         o => o.PercentDiscount == 0.0)
16                       .Collect()
17                       .Where(c => c.Any()));
18
19                 Then()
20                     .Do(ctx => ApplyDiscount(orders, 10.0))

```

```

20         .Do(ctx => ctx.UpdateAll(orders));
21     }
22
23     private static void ApplyDiscount(IEnumerable<
24         ↪ Order> orders, double discount)
25     {
26         foreach (var order in orders)
27         {
28             order.PercentDiscount = discount;
29         }
30     }

```

Tras aplicar el descuento, se envía la etiqueta con que el descuento ha sido aplicado:

```

1     public class DiscountNotificationRule : Rule
2     {
3         public override void Define()
4         {
5             Customer customer = default;
6
7             When()
8                 .Match<Customer>(() => customer)
9                 .Exists<Order>(o => o.Customer == customer
10                    ↪ , o => o.PercentDiscount > 0.0);
11
12             Then()
13                 .Do(_ => customer.NotifyAboutDiscount());
14         }
15     }

```

- Por último, se cargan las reglas según se desea en el repositorio de reglas, se compilan, se crea una sesión de trabajo y se insertan los objetos a la base de hechos, para luego lanzar las reglas.^[1]

```

1     //Cargamos las reglas
2     var repository = new RuleRepository();
3     repository.Load(x => x.From(typeof(
4         ↪ PreferredCustomerDiscountRule).Assembly));
5
6     //Compilamos las reglas
7     var factory = repository.Compile();
8
9     //Creamos la sesion
10    var session = factory.CreateSession();
11
12    //Creamos los modelos
13    var customer = new Customer("John Doe") {IsPreferred =
14        ↪ true};
15    var order1 = new Order(123456, customer, 2, 25.0);
16    var order2 = new Order(123457, customer, 1, 100.0);
17
18    //Insertamos los hechos a la base de hechos
19    session.Insert(customer);
20    session.Insert(order1);
21    session.Insert(order2);

```



```
20
21      //Lanzamos las reglas
22      session.Fire();
```

Aspectos relevantes del desarrollo del proyecto

5.1. Ciclo de vida utilizado

El ciclo de vida utilizado para el desarrollo de esta aplicación, que se ha procurado seguir durante todo el tiempo de desarrollo, es un ciclo de vida de tipo ágil. Se ha escogido este tipo de ciclo de vida puesto que permite tener el control de todos los aspectos de las diferentes fases de desarrollo de la aplicación, tanto a nivel de desarrollo como a nivel de documentación, permitiendo también la alternancia entre fases acorde con las necesidades del proyecto.

Un ciclo de vida ágil consiste en una entrega de forma incremental de las diferentes características de la aplicación y mejora de las características ya existentes de la misma. Durante este ciclo de vida se han ido subiendo diferentes modificaciones tanto a *GitHub* como a *Trello*, las cuales son denominadas como sprints. Cada sprint representa una actualización de una característica o conjunto de características de la aplicación o de los anexos, las cuales ayudan a marcar qué actividad se ha realizado y en qué momento se ha realizado.

En este caso, se ha utilizado un método visual de Kanban.[\[16\]](#)

Kanban es un método de gestión del flujo de trabajo para definir, gestionar y mejorar los servicios que proporciona el trabajo de conocimiento, siendo de gran ayuda para la visualización del trabajo, la optimización de la eficiencia y la continuación de diferentes mejoras. El trabajo se representa en tableros Kanban, lo que permite optimizar la entrega de trabajo a través

de múltiples equipos y manejar, incluso los proyectos más complejos en un solo entorno.

La implementación de este modelo de ciclo de vida el ha sido posible gracias a *Trello*, que es una herramienta muy útil para la gestión de proyectos a nivel visual en el que cual se tienen diferentes columnas sobre las actividades por hacer, las actividades en realización, las actividades realizadas y las diferentes reuniones de seguimiento que se han tenido con el tutor.

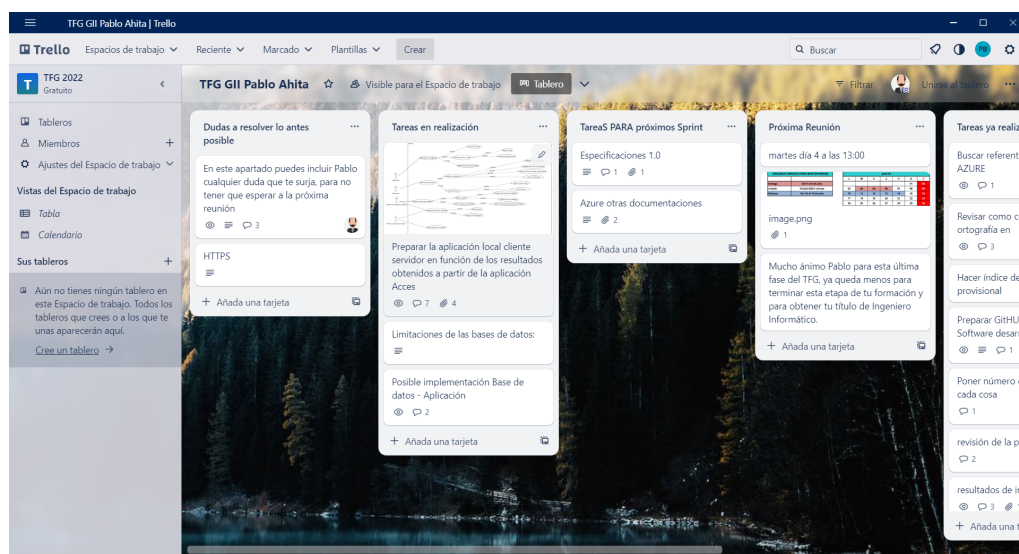


Figura 5.1: Uso de *Trello* para el proyecto

Como se puede comprobar en la siguiente captura, durante todo este tiempo se han colocado diferentes columnas en las que se han colocado diferentes aspectos:

- **Dudas a resolver lo antes posible:** En este apartado el alumno tiene la posibilidad de colocar esas inquietudes que tiene en cualquier aspecto relacionado con el proyecto sin necesidad de comentarlo explícitamente en cualquiera de las reuniones de seguimiento que se han ido realizando durante el curso.
- **Tareas en realización:** En este apartado se introducen las tareas en realización en ese mismo momento, las cuales se han ido marcando en cada una de las reuniones de seguimiento a lo largo de todo el tiempo de desarrollo.

- **Tareas para próximos sprint:** En este apartado se introducen las tareas que se realizarán para próximos sprints.
- **Próxima reunión:** En este apartado se introduce la información sobre la fecha de la próxima reunión y a la hora en la que tendrá lugar. Durante todo este tiempo las reuniones han tenido lugar los martes de cada semana por la mañana, ya sea a las 11:30 horas o a las 13:00 horas.
- **Tareas ya realizadas:** En este apartado se introducen todas las tareas que ya han sido realizadas durante el tiempo de desarrollo.
- **Información de las reuniones realizadas:** En este apartado se registran todas las reuniones realizadas durante todo este tiempo. Todas ellas han estado en el apartado anterior.

5.2. Fases de análisis, diseño e implementación

Para poder seguir el ciclo de vida anteriormente mencionado, se tienen que tener claras las fases a seguir durante todo el proceso de desarrollo. Al tratarse de un ciclo de vida ágil, cabe resaltar que el paso entre fases es cíclico puesto que, como se ha mencionado con anterioridad, conviene que el proceso de desarrollo de la aplicación se adapte a las necesidades que se vayan teniendo a lo largo del proyecto, por lo que las fases de *análisis*, *diseño* e *implementación* se van alternando a lo largo del tiempo.

Análisis

El análisis del entorno ha sido realizado con la ayuda principal de la aplicación en Microsoft Access, la cual ha servido como punto de referencia para montar las bases en las que se apoya dicha aplicación. En la aplicación Access ya había posibilidad de registrar organizaciones, de crear equipos evaluadores, de realizar test de indicadores y de guardar los registros de dichos test de indicadores en la base de datos en este mismo programa. Era un procedimiento bastante arcaico el uso de esa aplicación, por lo que se ha tenido que plantear que nuevas características se pretenden añadir en esta nueva versión:

- En primer lugar, la aplicación tiene que almacenar los usuarios mediante el registro de los mismos en la base datos, en lugar de hacerlo

directamente en el apartado de equipos evaluadores. Los usuarios se segregan en tres categorías diferentes:

- **Administrador:** Es el usuario que posee más permisos en cuanto a funcionalidades de la aplicación se refiere. Dicho actor será el director de *Fundación Miradas*, quien tendrá la potestad de manejar quiénes pueden acceder a la organización y qué organizaciones pueden acceder a ella, además de poder gestionar aquellas operaciones relacionadas con las evaluaciones de indicadores, relleno de los valores de las evidencias y calculando el valor total de cada indicador a partir de las mismas, también calculando el valor total obtenido mediante el valor total de cada indicador, operaciones que ejerce el sistema experto del lado del servidor.
- **Director de organización externa:** Este actor actúa como una especie de ayudante para el actor *Administrador*, puesto que tiene la potestad de añadir los equipos evaluadores y los centros de su organización, aparte de modificarlos y gestionarlos, con tal de no depender de *Fundación Miradas* para poder realizar esos cometidos.
- **Usuario de organización:** Es un usuario el cual únicamente puede ver los resultados de los test de indicadores a los que pertenece.

Se ha decidido segregar de esta forma a los usuarios por pura distinción de sus responsabilidades (*Para ver más detalles ir al capítulo de Requisitos en los anexos*), algo que en la aplicación antigua de Access no se tenía o no se necesitaba tener en cuenta en su momento.

- Al igual que los usuarios, la segregación de organizaciones es también un factor de vital importancia. Dicha segregación es en la que se basan los usuarios para determinar si el usuario es de la *Fundación Miradas* o por lo contrario si es un usuario de otra organización. Cada una de las organizaciones se identifican por **identificador de organización** de tipo entero, un **tipo de organización** que puede ser *EVALUATED* o *EVALUATOR* y un **trastorno**, que en nuestro caso siempre es *AUTISM*. Se ha decidido que se identifiquen de esta manera para que esta entidad pueda ser utilizada también para otro tipo de diagnósticos o enfermedades, requisito exigido por el tribunal antes de comenzar con el desarrollo. Por lo tanto, nos podemos encontrar con dos grandes tipos de organizaciones:

- **Organizaciones evaluadoras (*Fundación Miradas*):** Dichas organizaciones no pueden ser añadidas como tal desde la aplicación, ya que la propia *Fundación Miradas* es la encargada de añadir organizaciones evaluadas, como se ha relatado con anterioridad. Para poder añadir una se tendría que utilizar *cURL* o *Azure Studio*.
 - **Organizaciones evaluadas:** Son las organizaciones que se someten a los diferentes test de indicadores. En contraparte con el otro tipo de organizaciones, éstas sí que pueden ser añadidas a partir de la propia aplicación, mediante el formulario de agregación de organizaciones. (Véase *manual de usuario en los Anexos o vídeos incluidos en el repositorio*)
- También se tiene que tener el control de los equipos evaluadores por parte exclusiva de los usuarios de la *Fundación Miradas*, a diferencia de la aplicación en Access la cual no hace distinción, como se ha mencionado con anterioridad. La estructura de los equipos evaluadores no ha cambiado con respecto a Access.
 - La realización de las pruebas de indicadores también sigue la misma dinámica, con la diferencia de que la realización de los mismos es más intuitiva y rápida gracias al uso en pantallas táctiles.

Teniendo en cuenta esos aspectos, se ha realizado tanto el análisis de los casos de uso, el cual se encuentra en el capítulo de requisitos de los anexos, como el diagrama de clases.

En cuanto a la toma de decisiones, al principio del desarrollo se pretendía unificar todo bajo el mismo lenguaje *Java*, ya que cuento con la experiencia en este lenguaje que se ha ido proporcionando a lo largo de las asignaturas del grado. Para la comunicación con la base de datos se utilizaba *JDBC*, el cual ejecuta las consultas SQL desde el propio código. Durante gran parte del tiempo de desarrollo de la aplicación, se han ido encontrando diferentes dificultades para poder ejecutar dichas órdenes en la base de datos, puesto que todo aquello que requiera una conexión a internet desde una aplicación Android, requiere que sea ejecutado en segundo plano, ya sea mediante *java.util.concurrent*, mediante callbacks o mediante técnicas de programación concurrente. Ese inconveniente obligó a empezar a trabajar con las operaciones al servidor en segundo plano, puesto que en un hilo tiene que estar ejecutándose la aplicación y en otro tiene que estar ejecutándose la solicitud al servidor. También se pretendía montar el servicio web en *JAX-RS* para que pueda realizar esas operaciones en la base de datos mediante

JDBC. Aunque esta técnica funcionaba correctamente, el procesamiento de la respuesta en Azure era bastante más lento debido a que no está integrado en Azure, en comparación a *C#* y al framework *ASP.NET*, los cuales por su integración en Azure tiene unos tiempos de respuesta bastante más rápidos en comparación con su semejante en *Java*.

Diseño

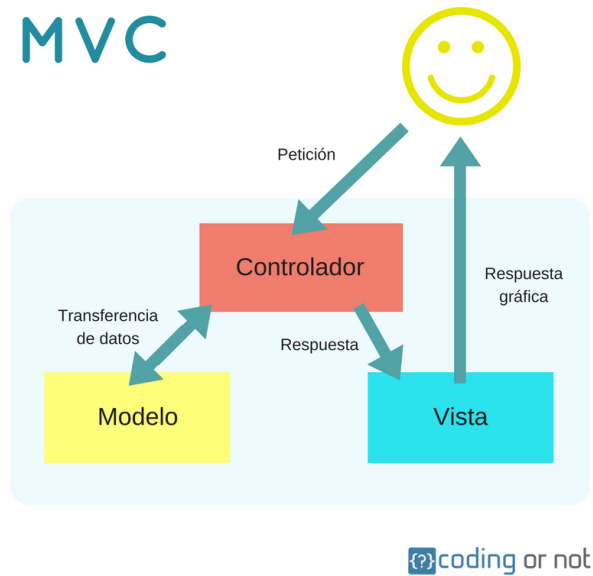


Figura 5.2: Estructura básica de la arquitectura *Modelo-Vista-Controlador*

En este caso se ha elegido como arquitectura el modelo denominado *Modelo-Vista-Controlador*. Dicho modelo segrega las diferentes responsabilidades que tienen los usuarios en los que cada una de las tres capas (*Modelo*, *Vista* y *Controlador*) tiene unas reglas exclusivas de ellas mismas.

- **Modelo:** El modelo se encarga de almacenar las diferentes entidades con las que interactúa el servidor, para poder realizar así las peticiones y poder devolver correctamente la respuesta al cliente, previa serialización en el cliente y previa deserialización en el servidor. En este caso el modelo consta de las clases *User*, *Organization*, *Indicator*, *Evidence*, *Center*, *Address*, *City*, *Province*, *Region*, *Country*, *EvaluatorTeam*, *IndicatorsEvaluation*, *IndicatorsEvaluationEvidencesReg*, *Indicator-sEvaluationIndicatorsReg* e *IndicatorsEvaluationSimpleEvidencesReg*

(Véase más información acerca de los modelos en el capítulo de Diseño de los Anexos). Cabe resaltar también que el modelo tiene que ser idéntico tanto en el cliente como en el servidor, para evitar así diferentes problemas en la serialización y deserialización de dichas entidades.

- **Vista:** La vista consiste en la forma en la que se muestran el contenido de las respuestas de las peticiones en la base de datos. También se incluye en este apartado la interfaz gráfica con la que interactúan los usuarios. En este caso, como cada tipo de usuario tiene responsabilidades diferentes, se considera que esta aplicación tiene diferentes vistas, todo por las responsabilidades que tienen los usuarios. En nuestro caso, la vista es proporcionada al cliente mediante las clases de tipo controlador o *Controller*, las cuales se encargan de enviar la solicitud al endpoint correspondiente del servidor y de procesar correctamente la respuesta que éste le da.
- **Controlador:** El controlador se encarga de realizar las peticiones a la base de datos previa solicitud del cliente. El controlador puede realizar todo tipo de operaciones *CRUD* (*Create, Read, Update, Delete*) de la base de datos, encargándose también de marcar los endpoints en los que se van a realizar dichas operaciones. Dichos métodos del controlador se apoyan en el contexto de base de datos de *Entity Framework* que realizan la consulta de la base de datos, todo ello en el lado del servidor.

Se ha optado por esta arquitectura debido a que se busca el control necesario para todo el proceso de las peticiones HTTP entre el cliente y el servidor. En primer lugar los modelos son los que se encargan de establecer la estructura JSON de las peticiones, de tal forma que en el proceso de serialización y de deserialización los campos sean asignados correctamente, para que la vista muestre las respuestas de la manera esperada de las peticiones de las operaciones realizadas por el controlador.

En cuanto a la base de datos se cuentan con las siguientes entidades SQL, se utiliza `CREATE TABLE` para poder crear las tablas, `INSERT INTO nombre_tabla VALUES` para modificar las tablas, `UPDATE nombre_tabla SET columna=valor WHERE columna=valor_old` para actualizar la columna, `SELECT * FROM nombre_tabla` para mostrar los registros de una tabla y `DELETE * FROM nombre_tabla` para hacer el delete de esa tabla. Para la información sobre la estructura de las entidades, véase el capítulo de diseño de los anexos.

Implementación

En cuanto a la implementación de la aplicación, se han ido encontrando diferentes problemas que se han ido solventando con el paso del tiempo, quedando algunas pendientes de resolver. Los errores que se han ido encontrando a lo largo del tiempo son los siguientes:

- **Android.NetworkOnMainThreadException:** Esta excepción ha saltado sobre todo al principio del proceso de desarrollo, en la que se estaba intentando conectar directamente la base de datos local a la aplicación sin intermediario alguno como el caso de un servidor. Esta excepción sale debido a que las operaciones de la base de datos y de comunicación pretenden hacerse en el mismo hilo en el que se ejecuta la aplicación, haciendo que esta se cierre repentinamente por esa incorrecta gestión de hilos. Para evitar que salte esta excepción en el cliente, se tiene que montar esa estructura *Modelo-Vista-Controlador* en el cliente vista con anterioridad, utilizando cualquier técnica de manejo de hilos, decantándome por herramientas de manejo de tareas en segundo plano, como *Future* de *java.util.concurrent*, por la razón mencionada en el capítulo anterior, donde se menciona la herramienta y las posibles alternativas a la misma.
- **400 Bad Request:** Esta excepción ha saltado debido a que el JSON no ha sido generado correctamente. Este fallo surge debido a un problema en la estructura del modelo o la serialización en el lado del cliente, ya que la forma en la que se envía el JSON no depende como tal del lado del servidor, que puede estar funcionando correctamente. Para evitar que salga este código de error, hay que asegurarse que en el cliente los campos se llamen exactamente igual que en el JSON que recibe el cliente, al igual que en el servidor y en la propia base de datos.
- **500 Internal Server Error:** Esta excepción surge por un problema interno en el servidor, por lo que el cliente no tiene por qué tener fallos en el modelo o en el procesamiento de la respuesta. Este fallo puede surgir por diferentes razones, por lo que es buena práctica hacer un debug en local del lado del servidor mediante una llamada al endpoint desde herramientas como *cURL*. Ejemplo:

```
1      curl -X POST -H "Content-Type: application/json" -d "
      ↪ {"idOrganization\:1,\"orgType\":"EVALUATED
      ↪ "\",\"illness\":"AUTISM\", \"idCenter\:1,\"
      ↪ centerDescription\":"Sede principal\", \"
      ↪ idAddress\:2,\"telephone\":"987654321}" https://
      ↪ localhost:7049/Centers
```

- Hay que considerar también que los puntos clave o breakpoints tienen que estar activados según la conveniencia que tenga el usuario en ese momento, así se puede observar paso a paso el motivo los pasos a seguir hasta llegar a la zona crítica. Este error puede surgir por sintaxis en la orden SQL y por violaciones de claves primarias y foráneas, al igual que por cualquier tipo de error que surja en el servidor.

Cambios realizados

Los cambios que se realizaron a partir del Trabajo de Final de Grado son los siguientes:

- Se ha dejado de utilizar `AsyncTask` debido a que es una clase deprecada desde la API 30[14], por lo que Google recomienda en su lugar el uso de las diferentes implementaciones de `java.util.concurrent`.
- Se ha implementado la opción de poder detener la evaluación de indicadores para poder almacenar los registros y poder continuar desde la último indicador revisado.
- Se han utilizado diferentes bibliotecas para poder generar los gráficos a dos vías: En primer lugar se han utilizado la biblioteca de diseño de Android para poder construir las tablas a partir de los elementos pre-diseñados de Android Studio, y por otro lado se ha usado la biblioteca `com.otaila.ZoomLayout`[4] para poder añadir la movilidad necesaria con posibilidad de ampliar o reducir el tamaño de la tabla. Se añade de la siguiente manera en layout:

```
1      <com.otaliastudios.zoom.ZoomLayout
2          android:layout_width="match_parent"
3          android:layout_height="match_parent"
4          android:scrollbars="vertical|horizontal"
5          app:transformation="centerInside"
6          app:transformationGravity="auto"
7          app:alignment="center"
8          app:overScrollHorizontal="true"
9          app:overScrollVertical="true"
10         app:overPinchable="true"
11         app:horizontalPanEnabled="true"
12         app:verticalPanEnabled="true"
13         app:zoomEnabled="true"
14         app:flingEnabled="true"
15         app:scrollEnabled="true"
16         app:oneFingerScrollEnabled="true"
17         app:twoFingersScrollEnabled="true"
18         app:threeFingersScrollEnabled="true"
19         app:minZoom="0.7"
20         app:minZoomType="zoom"
21         app:maxZoom="2.5"
```

```

22         app:maxZoomType="zoom"
23         app:animationDuration="280"
24         app:hasClickableChildren="false">
25
26         <!-- Content here. -->
27
28     </com.otaliastudios.zoom.ZoomLayout>

```

- Para la generación de informes se ha utilizado XWPFDocument de la biblioteca org.apache.poi[8]. Un ejemplo se puede apreciar en el siguiente fragmento de código:

```

1
2     try (InputStream is = FileManager.getInstance().
        ↪ downloadReportTemplate(current_evaluation.
        ↪ getEvaluationType()).join()) {
3
4     ;
5     // Abrir el documento Word
6     XWPFDocument document = null; // Obtencion del documento
        ↪ con el InputStream
7     try {
8         document = new XWPFDocument(is);
9
10        // Obtener todas las tablas en el documento
11        List<XWPFTable> tables = document.getTables(); //
        ↪ Obtener las tablas
12
13        XWPFTable indicatorsTable=tables.get(3); //
        ↪ Obtener la cuarta tabla
14
15        List<XWPFTableRow> rows4=indicatorsTable.
        ↪ getRows(); // Obtener las filas de la
        ↪ cuarta tabla
16
17        for(XWPFTableRow row:rows4){
18            List<XWPFTableCell> cells=row.
        ↪ getTableCells();
19            for(XWPFTableCell cell:cells){
20                try{
21                    int numIndicator=Integer.parseInt(
22                        ↪ cell.getText()); // Intenta
                        ↪ obtener el numero del
                        ↪ indicador
23                    if(numIndicator<=indicatorRegs.
        ↪ size()){
24                        String color="";
25                        //Asignamos color a partir del
        ↪ estado
26                        if(indicatorRegs.get(
        ↪ numIndicator-1).
        ↪ getStatus().equals("
        ↪ IN_START")){
27                            color="FF0000";
28                        } else if (indicatorRegs.get(
        ↪ numIndicator-1).

```

```

29         ↪ getStatus().equals("
30         ↪ IN_PROCESS")) {
31             color="FFFF00";
32     }else{
33         color="00FF00";
34     }
35     setCellFormat(cell, false,
36         ↪ color, ""); //Formateamos
37         ↪ la tabla
38     }
39     }catch(NumberFormatException e){
40         //Continuamos si no hay numero
41     }
42 }
43
44     catch (IOException e) {
45         e.printStackTrace();
46     }
47 }catch (IOException e) {
48     e.printStackTrace();
49 }

```

- Se ha cambiado la estructura del equipo evaluador para poder soportar más de cuatro fechas de evaluación, las cuales no necesitan introducirse manualmente gracias a la biblioteca `com.aminography`, y a las clases `PrimeCalendar`^[2], la cual se utiliza para almacenar las fechas y `PrimeDatePicker`^[3], la cual proporciona los selectores de varias fechas de forma libre. En primer lugar, para seleccionar una única fecha, se utiliza el método `pickSingleDay`:

```

1         creationDateEditText.setOnClickListener(new View.
2             ↪ OnClickListener() {
3                 @Override
4                 public void onClick(View v) {
5                     PrimeCalendar today = new CivilCalendar();
6                     ↪ //Dia de hoy en calendario
7                     ↪ gregoriano
8
9                     PrimeDatePicker datePicker =
10                        ↪ PrimeDatePicker.Companion.dialogWith
11                        ↪ (today)
12                        .pickSingleDay(new
13                            ↪ SingleDayPickCallback() {
14                                @Override
15                                public void onSingleDayPicked(
16                                    ↪ PrimeCalendar singleDay)
17                                    ↪ {
18                                    creationDate=singleDay;
19                                    creationDateEditText.
20                                        ↪ setText(creationDate
21                                        ↪ .getLongDateString()
22                                        ↪ .split(", ")[1]); //
23                                        ↪ Formateo de la fecha
24                                }
25                            }
26                )
27            }
28        )

```

```

13         })
14         .minPossibleDate(today)
15         .build();
16
17         datePicker.show(getSupportFragmentManager
18             ↪ (), "CREATION_DATE");
19     }
20 }
21 });

```

Y para seleccionar varias fechas, lo haríamos con `pickMultipleDays`:

```

1         evaluationDatesEditText.setOnClickListener(new View.
2             ↪ OnClickListener() {
3             @Override
4             public void onClick(View v) {
5                 PrimeCalendar today = new CivilCalendar();
6
7                 if(evaluationDates==null){
8                     evaluationDates=new ArrayList<>();
9                 }
10                PrimeDatePicker datePicker =
11                    ↪ PrimeDatePicker.Companion.dialogWith
12                    ↪ (today)
13                    .pickMultipleDays(new
14                        ↪ MultipleDaysPickCallback() {
15                        @Override
16                        public void
17                            ↪ onMultipleDaysPicked(
18                            ↪ List<PrimeCalendar> list
19                            ↪ ) {
20                            if(!evaluationDates.
21                                ↪ isEmpty()) {
22                                evaluationDates.clear
23                                ↪ ();
24                            }
25                            evaluationDates.addAll(
26                                ↪ list);
27                            String text="";
28                            Collections.sort(
29                                ↪ evaluationDates, new
30                                ↪ Comparator<
31                                ↪ PrimeCalendar>() {
32                                @Override
33                                public int compare(
34                                    ↪ PrimeCalendar o1
35                                    ↪ , PrimeCalendar
36                                    ↪ o2) {
37                                    if(o1.
38                                        ↪ getTimeInMillis
39                                        ↪ () < o2.
40                                        ↪ getTimeInMillis
41                                        ↪ ()){
42                                        return -1;
43                                    } else if(o1.
44                                        ↪ getTimeInMillis
45                                        ↪ () > o2.
46                                        ↪ getTimeInMillis
47                                        ↪ ()){
48                                        return 1;

```

```

25         }
26         return 0;
27     }
28 });
29 if(evaluationDates.size()
    ↪ >=MIN_NUM_EVAL_DATES
    ↪ && evaluationDates.
    ↪ get(evaluationDates.
    ↪ size()-1).
    ↪ getTimeInMillis()-
    ↪ evaluationDates.get
    ↪ (0).getTimeInMillis
    ↪ () < 2629800000L){
30     StringBuilder sb=new
    ↪ StringBuilder();
31     for(int i=0;i<
    ↪ evaluationDates.
    ↪ size();i++){
32         sb.append(
    ↪ evaluationDates
    ↪ .get(i).
    ↪ getLongDateString
    ↪ ().split(",
    ↪ ")[1]);
33         if(i<
    ↪ evaluationDates
    ↪ .size()-1){
34             sb.append(", "
    ↪ );
35         }
36     }
37     text=sb.toString();
38 }else{
39     if(!evaluationDates.
    ↪ isEmpty()) {
40         evaluationDates.
    ↪ clear();
41     }
42     String msg="";
43     if(evaluationDates.
    ↪ size()<
    ↪ MIN_NUM_EVAL_DATES
    ↪ ){
44         msg="<b>"+
    ↪ getString(R.
    ↪ string.
    ↪ must_select_three_eval_dates
    ↪ )+"</b>";
45     }
46     else if(
    ↪ evaluationDates.
    ↪ get(
    ↪ evaluationDates.
    ↪ size()-1).
    ↪ getTimeInMillis
    ↪ ()-
    ↪ evaluationDates.
    ↪ get(0).
    ↪ getTimeInMillis

```

```

47      ↪ () >= 2629800000L)
      ↪ {
      ↪     msg = "<b>" +
      ↪         getString(R.
      ↪         string.
      ↪         difference_between_dates_is
      ↪         ) + "</b>";
48  }
49  new AlertDialog.
      ↪ Builder(
      ↪ RegisterNewEvaluatorTeam
      ↪ .this)
50      .setTitle(
      ↪     getString
      ↪     (R.
      ↪     string.
      ↪     error))
51      .setMessage(
      ↪     Html.
      ↪     fromHtml
      ↪     (msg, 0))
52      .setIcon(
      ↪     android.
      ↪     R.
      ↪     drawable
      ↪     .
      ↪     ic_dialog_alert
      ↪     )
53      .
      ↪     setPositiveButton
      ↪     (
      ↪     getString
      ↪     (R.
      ↪     string.
      ↪     understood
      ↪     ), new
      ↪     DialogInterface
      ↪     .
      ↪     OnClickListener
      ↪     () {
54         @Override
55         public
      ↪         void
      ↪         ↪
      ↪         onClick
      ↪         (
      ↪         DialogInterface
      ↪         ↪
      ↪         dialog
      ↪         ,
      ↪         int
      ↪         which
      ↪         ) {
56             dialog
      ↪             ↪
      ↪             ↪ dismiss
      ↪             ↪ ()
      ↪             ↪ ;
      ↪             ↪

```



```

57                                     }
58                                     })
59                                     .create().show
59                                     ↪ ();
60
61                                     }
62                                     evaluationDatesEditText.
62                                     ↪ setText(text);
63
64                                     }
65                                     })
66                                     .initiallyPickedMultipleDays(
66                                     ↪ evaluationDates)
67                                     .minPossibleDate(today)
68                                     .build();
69                                     datePicker.show(getSupportFragmentManager
69                                     ↪ (), "EVALUATION_DATES");
70                                     }
71                                     });

```

- Se ha añadido la posibilidad de añadir fotos de perfil a usuarios, organizaciones, equipos evaluadores y centros. Para subir los ficheros a un contenedor de Azure, se ha creado el siguiente método en la clase `FileManager`:

```

1      public static void uploadFile(InputStream inputStream,
2      ↪ String containerName, String fileName){
3      Runnable task=()->{
4
5          // Get the BlobContainerClient;
5          containerClient = blobServiceClient.
5          ↪ getBlobContainerClient(containerName);
6
7
8          // Get the BlobClient
9          blobClient = containerClient.getBlobClient(
9          ↪ fileName);
10
11      try {
12          BinaryData data=BinaryData.fromStream(
12          ↪ inputStream);
13          blobClient.upload(data,true);
14      }catch(Throwable t){
15          if(!(t instanceof IllegalArgumentException)){
15          ↪ //Exception appears, but file uploads
15          ↪ correctly, obtaining a blob url
15          throw t;
16      }
17      }
18      };
19
20      task.run();
21  }
22

```

Como punto adicional, al inicio de sesión se descargan las fotos de perfil del usuario y de la organización de forma paralela. Para ello se

utiliza un `CountDownLatch` con tantos hilos como ficheros querramos descargar.

```

1      public static void downloadPhotosProfileAsync(List<
        ↳ String> fileNames, final PhotosDownloadCallback
        ↳ callback) {
2          // Contador para esperar a que todas las descargas se
        ↳ completen
3          CountDownLatch latch = new CountDownLatch(fileNames.
        ↳ size());
4          List<ByteArrayOutputStream> resultStreams = new
        ↳ ArrayList<>();
5
6          for (String fileName : fileNames) {
7              // Get the BlobContainerClient
8              BlobContainerClient containerClient =
        ↳ blobServiceClient.getBlobContainerClient("
        ↳ profile-photos");
9              // Get the BlobClient
10             BlobClient blobClient = containerClient.
        ↳ getBlobClient(fileName);
11
12             if(!fileName.isEmpty()) {
13                 // Ejecutar cada descarga en un hilo separado
14                 new Thread(() -> {
15                     ByteArrayOutputStream stream = new
        ↳ ByteArrayOutputStream();
16                     try {
17                         blobClient.downloadStream(stream);
18                         numAttempts = 0;
19                         // Llamar al callback en caso de exito
20                         callback.onPhotoDownloadSuccess(
        ↳ fileName, stream);
21                     } catch (Exception e) {
22                         if (e.getCause() instanceof
        ↳ SocketTimeoutException) {
23                             numAttempts++;
24                             if (numAttempts < 3) {
25                                 // Intentar nuevamente la
        ↳ descarga recursivamente
26                                 downloadPhotosProfileAsync(
        ↳ Collections.
        ↳ singletonList(fileName),
        ↳ callback);
27                             } else {
28                                 numAttempts = 0;
29                                 // Llamar al callback en caso
        ↳ de falla despues de
        ↳ varios intentos
30                                 callback.
        ↳ onPhotoDownloadFailure(
        ↳ fileName, new
        ↳ RuntimeException("Numero
        ↳ maximo de intentos
        ↳ alcanzado", e));
31                             }
32                         } else {
33                             // Llamar al callback en caso de
        ↳ otro tipo de error

```

```

34         callback.onPhotoDownloadFailure(
            ↪ fileName, new
            ↪ RuntimeException("Error en
            ↪ la descarga", e));
35     }
36 } finally {
37     latch.countDown(); // Reducir el
            ↪ contador del latch cuando una
            ↪ descarga se completa
38 }
39 }).start();
40 }else{
41 }
42 }
43 }
44
45 try {
46     latch.await(); // Esperar hasta que todas las
            ↪ descargas se completen
47 } catch (InterruptedException e) {
48     e.printStackTrace();
49 }
50
51 }

```

Y en la clase ProfilePhotoUtil, creamos un callback para asignar la fotografía al correspondiente bitmap.

```

1     private ProfilePhotoUtil(String profilePhotoUsr,String
    ↪ profilePhotoOrg){
2         this.profilePhotoUsr = profilePhotoUsr;
3         this.profilePhotoOrg = profilePhotoOrg;
4
5         List<String> fileNames=new ArrayList<>();
6         if(!profilePhotoUsr.isEmpty()) {
7             fileNames.add(profilePhotoUsr);
8         }
9         if(!profilePhotoOrg.isEmpty()) {
10            fileNames.add(profilePhotoOrg);
11        }
12
13        if(!fileNames.isEmpty()) {
14            // Descargar fotos en paralelo
15            FileManager.downloadPhotosProfileAsync(fileNames,
    ↪ new FileManager.PhotosDownloadCallback() {
16                @Override
17                public void onPhotoDownloadSuccess(String
    ↪ fileName, ByteArrayOutputStream stream)
    ↪ {
18                    if (fileName.equals(profilePhotoUsr)) {
19                        imgUser = getBitmapFromStream(stream);
20                    } else {
21                        imgOrg = getBitmapFromStream(stream);
22                    }
23                }
24            }
25            @Override

```

```

26         public void onPhotoDownloadFailure(String
           ↪ fileName, Exception e) {
27
28         }
29
30
31     });
32 }
33 }

```

- Los informes, al igual que las fotografías de perfil, son almacenados en un contenedor de Azure, utilizando un método de FileManager para dicho cometido:

```

1     public static CompletableFuture<ByteArrayOutputStream>
           ↪ downloadReport(String fileName){
2     return CompletableFuture.supplyAsync(() -> {
3         ByteArrayOutputStream stream = new
           ↪ ByteArrayOutputStream();
4         // Get the BlobContainerClient
5         BlobContainerClient containerClient =
           ↪ blobServiceClient.getBlobContainerClient("
           ↪ reports");
6         // Get the BlobClient
7         BlobClient blobClient = containerClient.
           ↪ getBlobClient(fileName);
8         try {
9             blobClient.downloadStream(stream);
10            numAttempts=0;
11        } catch (Exception e) {
12            if(e.getCause() instanceof
           ↪ SocketTimeoutException){
13                numAttempts++;
14                if(numAttempts<3) {
15                    return downloadReport(fileName).join()
           ↪ ;
16                }
17                else{
18                    numAttempts=0;
19                    return null;
20                }
21            }
22            else{
23                throw new RuntimeException(e);
24            }
25        }
26        return stream;
27    });
28 }

```

Posteriormente, en el método donde se llame, se añade el método `join()` de `CompletableFuture`:

```

1     FileManager.downloadReport(fileName).join();

```

- Se han añadido actividades en la aplicación para editar usuarios, organizaciones, equipos evaluadores y centros.
- Se ha ampliado la precarga geográfica (para los desplegables) a todos los países de Hispanoamérica y de Portugal.
- Los campos de número de teléfono han pasado de ser `Long` o `BIGINT` a ser `String` o `VARCHAR`, puesto que añadimos el prefijo telefónico de cada país.
- En las evaluaciones de indicadores, se permite la adición de conclusiones al finalizar la evaluación de indicadores.
- En las evaluaciones de indicadores, se permite la adición de oportunidades de mejora, ya sea para la generalidad del indicador o para cada una de las cuatro evidencias, ayudando a determinar por qué no se cumple una evidencia o no se ha alcanzado un indicador.
- Se ha creado una web app sencilla que ayuda al administrador a gestionar los registros, con la finalidad de que cualquier usuario no pueda registrarse sin previa autorización de este actor. Dicha web app[5] garantiza la seguridad que necesita una operación de tal importancia.
- Se ha implementado la implementación continua de GitHub en Azure, la cual permite que en cada commit se implemente la web app del lado del servidor.
- Se han creado layouts específicos para tablet y para pantallas horizontales, al igual que bloquear la orientación en la mayoría de actividades.
- Se han añadido apartados de ayuda en actividades clave de la aplicación, ayudando a que el usuario tenga mayor intuitividad en el uso de la misma.
- Se ha creado una web app para poder descargar los informes desde el ordenador.[6]

Funcionalidades descartadas

Aparte de eso, se han descartado otras funcionalidades adicionales por su falta de necesidad prioritaria:

- En primer lugar, se ha desechado la funcionalidad de editar, añadir y eliminar indicadores, evidencias, ámbitos y subdivisiones, ya que

dicho contenido no es variable a lo largo del tiempo, sino que dicho contenido es fijo y no pretende cambiarse por el momento, al tratarse de una referencia clara para la mejora de la calidad de vida de las personas con trastorno del espectro autista.

- Aunque se haya implementado un sistema de traducción en el lado del servidor, y habiendo colocado diez columnas para cada idioma, es una API muy limitada, por lo que se ha desechado su uso porque supondría un sobre coste adicional por dicho servicio, ya sea a través de Azure o a través de otros servicios. Además, el mercado del proyecto va a ser hispanohablante, por lo que no es una funcionalidad estrictamente necesaria en este momento.
- De igual manera, se ha desechado la opción de implementar un servicio de correo electrónico y de SMS para la gestión de registros de usuario, ya que de igual manera el coste de dichos servicios es muy elevado, además de que bajo el contexto de funcionamiento del trabajo de *Fundación Miradas*, la comunicación constante es primordial para la notificación de cualquier novedad.
- Se ha eliminado la clase `EvaluatorTeamMember`, ya que era innecesaria para poder agregar a los usuarios de los equipos evaluadores.
- Se ha desechado la pantalla de *Actividad reciente* y se ha sustituido con un menú principal personalizado para cada tipo de usuario, favoreciendo la intuitividad de la aplicación.
- Se han generado los gráficos de manera manual mediante bibliotecas de Java, al igual que el uso de `XWPFDocument` con plantillas para la generación de las tablas, en lugar de realizar los gráficos mediante Azure.
- Si utilizamos el campo `passwordUser` de la entidad `User` como internal, no se puede acceder a él dentro de la serialización en cliente, por lo que en cliente se utiliza `JsonDocument (C#)` y `JsonObject (Java)` para los getters de usuario, sin mostrar la contraseña.
- Se ha desechado el uso de un menú de preferencias, ya que no es estrictamente necesaria. Para solventarlo, se han utilizado pictogramas en las imágenes y se ha mejorado la visualización de la interfaz para que no sea tan necesario.
- Es muy complicado hacer que *Apple* acepte un proyecto de código abierto, por lo que se ha desechado por el momento, por lo que los

sistemas **Android**, siendo más económicos que los dispositivos **Apple**, son la opción principal para el desarrollo de la aplicación. Aun eso, ciertas funcionalidades se utilizan mediante web apps.

Internacionalización

La internacionalización de una aplicación es un aspecto clave para garantizar su accesibilidad y usabilidad en diferentes países y culturas. En el caso de una aplicación de Android Studio, es posible implementar la internacionalización de manera efectiva utilizando diferentes recursos proporcionados por la plataforma. A continuación, se presenta un apartado sobre la internacionalización de la aplicación, considerando los idiomas español, inglés, francés, euskera, catalán, neerlandés, gallego, alemán, italiano y portugués, y la posibilidad de ampliarlo a otros idiomas en el futuro:

- **Internacionalización de la interfaz gráfica:** La interfaz gráfica de la aplicación se tradujo el curso pasado a tres idiomas: español, inglés y francés. Este año se han añadido siete idiomas más a la interfaz: euskera, catalán, gallego, neerlandés, alemán, italiano y portugués. Android Studio proporciona una herramienta llamada *Translations Editor* que facilita la gestión de los ficheros strings.xml para cada idioma. Esta herramienta permite configurar de manera interactiva los textos de la interfaz en los diferentes idiomas. Para implementar la internacionalización de la interfaz gráfica en la aplicación, se deben seguir los siguientes pasos:
 - *Creación de ficheros strings.xml:* Para cada idioma, se deben crear ficheros strings.xml separados en diferentes directorios. Por ejemplo: strings.xml para el idioma por defecto (inglés) en el directorio *values*, strings.xml para el español en el directorio *values-es* y strings.xml para el francés en el directorio *values-fr*, así para los diez idiomas disponibles.
 - *Definición de cadenas de texto:* En cada fichero strings.xml, se deben definir las cadenas de texto utilizadas en la interfaz gráfica de la aplicación. Cada cadena debe tener un identificador único y su correspondiente traducción en el idioma correspondiente.
 - *Acceso a las cadenas de texto en el código:* Para mostrar las cadenas de texto en la interfaz, se debe acceder a ellas mediante su identificador en el código de la aplicación. Por ejemplo, utilizando `getString(R.string.mi_cadena)`.

Con estos pasos, la interfaz gráfica de la aplicación estará internacionalizada y mostrará los textos correspondientes al idioma configurado en el dispositivo siempre y cuando éste esté disponible.

- **Internacionalización de los datos obtenidos de la base de datos:** Además de la interfaz gráfica, es importante internacionalizar los datos obtenidos de la base de datos de la aplicación. En este caso, se ha optado por utilizar la biblioteca *Locale* de Java para detectar el idioma del dispositivo y mostrar los indicadores y evidencias en el idioma correspondiente. Para implementar la internacionalización de los datos, se pueden seguir los siguientes pasos:

- *Obtención del idioma del dispositivo:* Utilizando la clase *Locale* de Java, se puede obtener el idioma configurado en el dispositivo. Por ejemplo: `Locale.getDefault().getLanguage()`.
- *Obtención de los datos de la base de datos:* Al obtener los datos de indicadores y evidencias de la base de datos, se deben considerar las traducciones correspondientes para cada idioma. Por ejemplo, si el idioma del dispositivo es español, se deben obtener los datos en español; si es francés, se deben obtener en francés, así con cada uno de los idiomas mencionados. El idioma predefinido en caso de no estar registrado en la aplicación es el idioma inglés.
- *Visualización de los datos en la interfaz:* Una vez obtenidos los datos en el idioma correspondiente, se pueden mostrar en la interfaz de la aplicación según el diseño y la estructura definida.

Es importante tener en cuenta que la implementación de la internacionalización para los datos de la base de datos puede variar según la estructura y la forma en que se accede a los datos en la aplicación. La biblioteca *Locale* de Java proporciona diferentes métodos y opciones para adaptar la aplicación al idioma del dispositivo.

5.3. Pruebas

Las pruebas realizadas con la aplicación suponen la comprobación de que todos los aspectos de la misma funcionan correctamente, sobre todo desde el punto de la funcionalidad básica de la misma.

Organización evaluadora creada

A la Fundación Miradas y a su administrador se les ha añadido desde Azure Data Studio, ya que la propia aplicación no permite la creación de usuarios y organizaciones de tan altos privilegios:

- En primer lugar se ha añadido a la *Fundación Miradas* como organización:
 - **Identificador de organización:** 1
 - **Tipo de organización:** Evaluadora
 - **Trastorno/enfermedad:** Autismo
 - **Nombre:** Fundación Miradas
 - **Dirección:** Calle Valdenúñez, 8, Burgos.
 - **Email:** fmiradas@fundacionmiradas.org
 - **Teléfono:** +34 622434974
- Posteriormente se ha creado una cuenta de administrador de Fundación Miradas para poder realizar las operaciones a partir de la aplicación:
 - **Nombre:** Miguel
 - **Apellidos:** Gómez Gentil
 - **Tipo de usuario:** Administrador
 - **Email:** fmiradas@fundacionmiradas.org
 - **Teléfono:** +34 654545454
 - **Identificador de organización:** 1
 - **Tipo de organización:** Evaluadora
 - **Trastorno/enfermedad:** Autismo

A partir de estos datos de base, podemos crear ejemplo de uso real.

Organización evaluada creada

Para crear una organización, se ha tenido que crear tanto a la propia organización como a su director:

- En primer lugar se han introducido los siguientes datos para la organización:

- **Identificador de organización:** 1
 - **Tipo de organización:** Evaluada
 - **Trastorno/enfermedad:** Autismo
 - **Nombre:** Españita Power
 - **Dirección:** Calle Fernando Alonso, 33, Oviedo.
 - **Email:** espanitapower@hotmail.com
 - **Teléfono:** +34 654545454
- Posteriormente, tras añadir el email del director de la organización, este tiene los siguientes datos:
 - **Nombre:** Pablo
 - **Apellidos:** Ahíta del Barrio
 - **Tipo de usuario:** Director
 - **Email:** pablete@hotmail.com
 - **Teléfono:** +34 654545454
 - **Identificador de organización:** 1
 - **Tipo de organización:** Evaluada
 - **Trastorno/enfermedad:** Autismo

Usuario creado

Adicionalmente a eso, se ha creado un usuario adicional para la organización *Españita Power*, el cual tiene los siguientes datos:

- **Nombre:** Luisito
- **Apellidos:** Comunica
- **Tipo de usuario:** Organización
- **Email:** luisito@hotmail.com
- **Teléfono:** +52 5521123456
- **Identificador de organización:** 1
- **Tipo de organización:** Evaluada
- **Trastorno/enfermedad:** Autismo

La finalidad es tener dos usuarios, el director y un usuario de organización, para poder asignar al equipo evaluador de turno.

Equipo evaluador creado

Para el equipo evaluador de prueba, se han introducido los siguientes datos:

- **Identificador de equipo evaluador:** 1
- **Fecha de creación:** 16 de mayo de 2024
- **Consultor externo:** Flavio Briatore
- **Email del responsable:** pablete@hotmail.com
- **Email del profesional de atención directa:** luisito@hotmail.com
- **Otros miembros:** Lawrence Stroll, Matt Watson, Mikey Brown, Dan Fallows
- **Identificador de organización evaluadora:** 1
- **Tipo de organización evaluadora:** Evaluadora
- **Identificador de organización evaluada:** 1
- **Tipo de organización evaluada:** Evaluada
- **Identificador de centro:** 1
- **Trastorno/enfermedad:** Autismo
- **Nombre de la persona con TEA:** Esteban Ocon
- **Nombre del familiar de la persona con TEA:** Fernando Alonso
- **Fechas de evaluación:** 18 de mayo de 2024, 19 de mayo de 2024 y 20 de mayo de 2024

Evaluación de indicadores creada

Con todo lo necesario creado, la parte más importante se pone a prueba con una evaluación de tipo completo, la cual, como hemos mencionado con anterioridad, consta de 70 indicadores con 4 evidencias como máximo a seleccionar.

La intención de dicha prueba es marcar los indicadores de la siguiente manera:

- Los indicadores del 1 al 30 deben de ser de estado **REACHED**, los cuales deben tener las cuatro evidencias marcadas.
- Los indicadores del 31 al 60 deben de ser de estado **IN_PROCESS**, los cuales deben tener o dos o tres evidencias marcadas.
- Los indicadores del 61 al 70 deben de ser de estado **IN_START**, los cuales deben tener una o ninguna evidencia marcada.

Dicha prueba garantiza que se cumplan todas las posibilidades dentro de la evaluación de indicadores, además de poner en marcha el sistema experto anteriormente mencionado y generar el informe para que se almacene en la nube.

La prueba se fue guardando en tres fases, las cuales son las mismas que se han mencionado con anterioridad para el marcado de las evidencias de los indicadores, con la finalidad de que se puedan guardar y después obtener gracias al almacenaje de los registros de indicadores y evidencias (*Véase Anexos*).

Tras haber realizado dicha prueba de indicadores, se han obtenido los siguientes resultados:

Nivel		Nº de indicadores anotados	Multiplicar por	
De interés fundamental		9	5	45
		11	4	44
		1	0	0
Interés alto		10	4	40
		6	3	18
		0	0	0
Interés medio		7	3	21
		6	2	12
		5	0	0
Menor interés		4	2	8
		7	1	7
		4	0	0
TOTAL				195

Figura 5.3: Resultados de prueba de indicadores de prueba

Como se puede comprobar, se ha obtenido una puntuación de 195 puntos, lo cual corresponde a una puntuación de nivel *Muy bueno*. En dicha tabla, obtenida a partir del informe generado, se comprueba el número de indicadores marcados con cada combinación de nivel de interés-estado de indicador, los cuales se multiplican por un multiplicador fijo para obtener ese resultado. La suma de los resultados de cada combinación es la puntuación total de la evaluación del indicadores.

Trabajos relacionados

Los trabajos final de grado de referencia que se han utilizado son los siguientes:

- **Prueba de concepto *Azure Monitor*:** Es un Trabajo Final de Grado que trata sobre aplicaciones APM (*Application Performance Monitor*), que son herramientas que diagnostican el rendimiento de las aplicaciones, con la finalidad de encontrar fallos en el programa, cuellos de botella, e incluso poder solucionarlos, evitando degradaciones. La funcionalidad utilizada en este TFG es *Azure Monitor*, el cual es la herramienta de este estilo proporcionada por Microsoft, cuyo objetivo es averiguar si el proyecto puede ser utilizado por una organización que tiene dos entornos completamente diferentes: un entorno de nube *Azure* donde se encuentra su sitio web y que podrá albergar en el futuro alguno de los servicios que ofrece, y otro entorno en el sitio donde residen el resto de los servicios.
- **Desarrollo de una plataforma de tratamiento y streaming de vídeo para difusión de la cultura utilizando instancias de *Azure*:** Es un Trabajo Final de Grado que trata sobre una herramienta que es utilizada por artistas de cine independientes para darse a conocer a través de publicidad en diferentes dispositivos, ya sea a través de aplicaciones móviles o a través de navegadores. *Azure* es utilizado como soporte para la base de datos y la posterior gestión de transacciones, lo que se busca para el proyecto de la Fundación Miradas.
- **Servicios en la nube con *Microsoft Azure* : desarrollo y operación de una aplicación Android con DevOps:** Este proyecto

trata sobre el desarrollo de una aplicación de Android con almacenamiento en la nube, utilizándose para el almacenamiento, visualización y procesamiento de fotografías. Dicho proyecto utiliza *Azure* para desplegar la aplicación en la nube, con ayuda de DevOps para poder alargar el ciclo de vida de la misma. Puede ser de utilidad también para poder desarrollar la aplicación tanto a nivel de frontend como a nivel de backend.

- **JIZT. Generación de resúmenes abstractivos en la nube mediante Inteligencia Artificial:** JIZT es un servicio de generación automática de resúmenes basado en la corriente *Cloud Native*, que se basa en los principios de los sistemas escalables, elasticidad y agilidad. Dicho servicio es sustentado por una arquitectura de micro-servicios dirigido por eventos, garantizando la alta disponibilidad del servicio, aparte de los tres principios mencionados con anterioridad. Dicha aplicación es multiplataforma, por lo que consume la API REST del servicio en la nube, donde cualquier usuario dispone de los resúmenes que desee.
- **UBUNurse:** Este proyecto consiste en una aplicación multi-dispositivo el cual almacena en la nube registros sobre la evaluación de la atención domiciliar por parte de dicho personal hacia un determinado paciente. El procedimiento a seguir consiste en elegir un paciente en la lista de la cual dispone el enfermero, posteriormente se elige el test a realizar y por último se obtienen los resultados de la realización de dicho test. Este software pretende automatizar dicho proceso para mejorar la eficiencia del personal sanitario y también para mejorar el proceso de evaluación de cada paciente.
- **Machine learning mediante Microsoft Azure: una aplicación sobre real-state:** En este proyecto se trata más a fondo las herramientas de las cuales dispone *Azure*, las cuales son utilizadas para la creación de elementos de machine learning, los cuales explican el funcionamiento de *Azure* con fines estadísticos. Se utiliza en dicho proyecto *Azure Machine Learning* con los datos empíricos de una inmobiliaria estadounidense para la predicción y posterior clasificación del valor de las viviendas, comparando posteriormente con otros modelos de clasificación y de regresión. A pesar de no ser un proyecto de bases de datos, se le da mucho hincapié a los gráficos y a la muestra de resultados, lo que puede servir de utilidad para la parte final de muestra de resultados.

- **Desarrollo de un Bot en la plataforma *Azure* para ayudar en el aprendizaje del lenguaje de programación C:** Este proyecto consiste en el desarrollo de un bot mediante *Azure* para ayudar al alumnado de 1º curso del Grado en Ingeniería Electrónica y Automática de la Universidad Politécnica de Cartagena al aprendizaje del lenguaje de programación C. Aparte de eso trata también sobre las herramientas de Inteligencia Artificial de *Azure*, como *LUIS (Language Understanding)*, aparte de que el software de dicho bot tiene soporte de incorporación de diferentes idiomas, lo que es un factor de gran importancia para el desarrollo de la app. Se ha escogido este trabajo de final de grado puesto que se trata de un proyecto bastante completo en cuanto a contenido a sacar de él y en cuanto a estructuración del contenido del mismo.

Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Por lo tanto las posibles ampliaciones que pueden realizarse en el futuro, aunque no sean estrictamente necesarias en el corto plazo, son las siguientes:

- Se puede ampliar la web app para poder pasar las funcionalidades de la app Android a una web app, como la de administración.
- Se puede ampliar aún más la internacionalización, ya sea ampliando más idiomas o haciendo un sistema de traducción.

En cuanto al propio proyecto en sí, ha sido un proyecto que me ha llenado muchísimo, ya que por mi discapacidad cognitiva siempre he estado muy concientizado sobre todos los aspectos que las personas con cualquier discapacidad, en este caso las personas con TEA, por lo que es una aplicación hecha para ayudar a las personas con discapacidad por parte de una persona con una discapacidad reconocida.

A la par dicho proyecto ha supuesto un gran aprendizaje para mí en todos los aspectos. En primer lugar en el aspecto laboral o académico, ya que

he tenido que refrescar conocimiento sobre algunas herramientas las cuales llevaba bastante tiempo sin utilizar y también he tenido que aprender a manejar herramientas las cuales desconocía en muy poco tiempo, como es el manejo de *Visual Studio 2022* con el lenguaje de programación *C#* y el framework *ASP.NET*, los cuales no se han impartido en las asignaturas de este grado, por lo que ha supuesto un aporte de conocimientos adicional para la entrada al mundo laboral. Posteriormente también está suponiendo un aprendizaje importante a nivel personal, ya que ha habido bastantes momentos difíciles producto de las diferentes dificultades que han ido surgiendo durante el tiempo de desarrollo del mismo los cuales han sido un desafío a nivel personal bastante grande, teniendo la fortuna de haber recibido el apoyo de mucha gente, como menciono en la parte de agradecimientos de esta memoria, lo cual ha supuesto un impulso muy grande para seguir peleando para sacar este proyecto adelante de la mejor forma posible y a su vez afrontar los diferentes desafíos que un Ingeniero Informático debe cumplir en el día a día y afrontarlos de forma tranquila y serena.

Bibliografía

- [1] Getting Started | NRules — nrules.net. <https://nrules.net/articles/getting-started.html?tabs=cli>.
- [2] GitHub - aminography/PrimeCalendar: PrimeCalendar provides all of the java.util.Calendar functionalities for Persian, Hijri, and ... dates. It is also possible to convert dates to each other. — github.com. <https://github.com/aminography/PrimeCalendar>. [Accessed 07-07-2024].
- [3] GitHub - aminography/PrimeDatePicker: PrimeDatePicker is a tool that provides picking a single day, multiple days, and a range of days. — github.com. <https://github.com/aminography/PrimeDatePicker>.
- [4] GitHub - natario1/ZoomLayout: 2D zoom and pan behavior for View hierarchies, images, video streams, and much more, written in Kotlin for Android. — github.com. <https://github.com/natario1/ZoomLayout>.
- [5] Herramienta de administración de usuarios de Guía OTEA — guiaoteaadmin.azurewebsites.net. <https://guiaoteaadmin.azurewebsites.net/>.
- [6] Herramienta de descarga de informes — guiaoteainformes.azurewebsites.net. <https://guiaoteainformes.azurewebsites.net/>.
- [7] Rules Engine for .NET | NRules — nrules.net. <https://nrules.net/?tabs=cli>.

- [8] XWPFDocument (POI API Documentation) — poi.apache.org. <https://poi.apache.org/apidocs/dev/org/apache/poi/xwpf/usermodel/XWPFDocument.html>.
- [9] Universidad de Burgos. Azure for education - universidad de burgos, 2020. [Página de acceso al enlace de Azure for Education].
- [10] Pablo Ahíta del Barrio. Guía OTEA. <https://github.com/pabloahita/guiaOTEA>, 2024.
- [11] Pablo Ahíta del Barrio. Trabajo de Final de Máster, Curso 2023-2024, Pablo Ahíta del Barrio, Vídeo de Demostración — youtube.com. <https://www.youtube.com/watch?v=VOENUdTqxTA>, 2024.
- [12] Pablo Ahíta del Barrio. Trabajo de Final de Máster, Curso 2023-2024, Pablo Ahíta del Barrio, Vídeo de Presentación — youtube.com. <https://www.youtube.com/watch?v=T2vsFoT4b0Y>, 2024.
- [13] J.L.C. Gómez. *Trastornos del espectro autista y calidad de vida: guía de indicadores para evaluar organizaciones y servicios*. Aula abierta / Editorial La Muralla. Editorial La Muralla, 2009.
- [14] Google. AsyncTask.
- [15] https://www.innovaciondigital360.com/periodista/equipo_editorial/. Sistemas expertos: Guía completa actualizada — innovaciondigital360.com. <https://www.innovaciondigital360.com/ia/sistemas-expertos-que-son-su-clasificacion-como-funcionan-y-para-que-se-utilizan/>.
- [16] kanbanize.es. ¿qué es kanban? explicación para principiantes, 2023. [¿Qué es Kanban? Explicación para principiantes].
- [17] Microsoft Learn. Características de visual studio, 2023. [Características de Visual Studio].
- [18] Microsoft Learn. Información sobre asp.net core, 2023. [Información sobre ASP.NET Core].
- [19] Microsoft Learn. ¿qué es visual studio?, 2023. [¿Qué es Visual Studio?].
- [20] Microsoft. Azure for education, 2023. [Página principal de Azure for Education].
- [21] Microsoft. Azure sql, 2023. [Azure SQL].

- [22] Microsoft. Microsoft azure, 2023. [Página principal de Microsoft Azure].
- [23] Microsoft. Microsoft learn, 2023. [Página oficial de Microsoft Learn].
- [24] Wikipedia. Android studio, 2023. [Artículo de Wikipedia sobre Android Studio].
- [25] Wikipedia. Java, 2023. [Artículo de Wikipedia sobre C Sharp].