



UNIVERSIDAD

REY JUAN CARLOS

**ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA
INGENIERÍA TÉCNICA EN INFORMÁTICA DE SISTEMAS**

Curso Académico 2009/2010

Proyecto de Fin de Carrera

**AGENDA COMUNICATIVA PARA PERSONAS
CON TGD O AUTISMO MEDIANTE
DISPOSITIVOS MÓVILES**

Autor: Miguel A. Santiago Manzano

Tutora: Estefanía Martín Barroso



AGRADECIMIENTOS

Toda etapa requiere una reflexión e interiorización, en este caso, el proyecto fin de carrera supone la culminación de unos estudios universitarios los cuales, sin duda, me han aportado una serie de destrezas técnicas, capacidad de reflexión, de abstracción y sobre todo tenacidad para afrontar el día a día de mi corta carrera profesional y por qué no, dentro del ámbito personal.

Por todo esto, aunque me acuerdo de muchas personas que me han acompañado en este trayecto, sólo mencionaré a mi madre, Cati Manzano Martín, por todo lo que ha aguantado, sobre todo, por mis constantes incursiones laborales durante la etapa universitaria.

En cuanto al proyecto fin de carrera en sí mismo, agradecer a Jebel y Vero, por exponerme la posibilidad de realizar este proyecto e ir de la mano conmigo en los primeros pasos del mismo. Laura y Leticia por su ayuda en todo lo relativo al autismo y por supuesto a la tutora, Estefanía, por su cercanía y por resolver todas las dudas que han ido surgiendo incluso en horario intempestivo.

Espero que disfruten de la lectura y os animo a probar la aplicación.



RESUMEN

El autismo es un trastorno del desarrollo permanente que impacta en el desarrollo normal del cerebro dentro de las áreas relacionadas con la interacción social, las habilidades comunicativas y la organización de las actividades de la vida diaria de las personas que padecen este trastorno. En concreto, aplicaciones desarrolladas que faciliten las labores de comunicación de la persona autista con su entorno y la organización de las tareas rutinarias a través de dispositivos móviles, pueden ayudar a solucionar estos problemas mejorando de esta forma la calidad de vida de la persona autista y de las personas encargadas de su cuidado.

Por este motivo, sería útil disponer de una agenda en la que se puedan programar tareas semanales, consultadas posteriormente por las personas que padecen autismo o TGD. Éste es el principal objetivo de este proyecto fin de carrera titulado “*Agenda Comunicativa para personas con TGD o autismo mediante dispositivos móviles*”, que consiste en el desarrollo de una aplicación para dispositivos móviles, en la cual se muestra una agenda semanal de tareas previamente programadas por el responsable de la persona autista.

La aplicación desarrollada se encuentra dividida en dos módulos: una herramienta de autor donde el responsable puede configurar fácilmente las tareas semanales de la persona autista y la agenda de tareas adaptada a la personas con TGD. Esta agenda permite una consulta rápida, fácil e intuitiva de las tareas programadas por parte del autista. La interfaz de la agenda se ha realizado de la forma más sencilla posible para minimizar tanto las opciones mostradas como las interacciones que tiene que realizar la persona autista con el dispositivo móvil.

El usuario final podrá consultar actividades que tiene que realizar cada día de la semana mostrándose en forma de imágenes descriptivas de la actividad en cuestión. Además se le facilitará una ayuda en formato audio que se reproducirá en el momento que el usuario pulse sobre la imagen. Esta ayuda será una descripción breve y concisa de la actividad a realizar. Las tareas serán previamente configuradas por el padre, tutor o responsable a través de la herramienta de autor de la aplicación. Esta herramienta provee el interfaz necesario para este cometido, diferenciado del interfaz de consulta de la tarea adaptado para personas con TGD.



TABLA DE CONTENIDOS

| | |
|--|-----------|
| 1. MOTIVACIÓN..... | 1 |
| 2. OBJETIVOS..... | 7 |
| 2.1 Descripción del problema..... | 7 |
| 2.2 Requisitos de usuario..... | 7 |
| 2.3 Estudio de alternativas..... | 8 |
| 2.4 Metodología empleada..... | 13 |
| 3. DESCRIPCIÓN INFORMÁTICA..... | 15 |
| 3.1 Casos de uso..... | 15 |
| 3.1.1 Diagramas casos de uso..... | 15 |
| 3.1.1 Descripción casos de uso..... | 18 |
| 3.2 Diagrama de actividad..... | 26 |
| 3.2.1 Diagrama de actividad para un usuario con el rol persona con autismo..... | 26 |
| 3.2.2 Diagrama de actividad para un usuario con el rol padre, responsable o tutor..... | 27 |
| 3.3 Diseño de bocetos de las pantallas de la aplicación..... | 29 |
| 3.4 Diseño de la base de datos..... | 33 |
| 3.5 Implementación..... | 35 |
| 3.5.1 Estructura..... | 35 |
| 3.5.2 Implementación..... | 36 |
| 3.5.3 Interfaz de usuario..... | 38 |
| 3.6 Escenarios..... | 47 |
| 3.6.1 Escenarios con el rol padre, responsable o tutor..... | 47 |
| 3.6.2 Escenarios con el rol persona con autismo o TGD..... | 53 |
| 4. CONCLUSIONES Y POSIBLES AMPLIACIONES..... | 55 |
| 5. BIBLIOGRAFÍA..... | 57 |
| ANEXO: CONFIGURACIÓN ENTORNO DE DESARROLLO..... | 59 |



ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1: Ejemplo de ayudas visuales en el programa <i>BoardMaker</i> | 4 |
| Figura 2: Estructura de Android..... | 10 |
| Figura 3: Esquema MVC..... | 13 |
| Figura 4: Esquema MVC de la aplicación “ <i>Agenda Comunicativa</i> ”..... | 14 |
| Figura 5: Casos de uso de un usuario autista | 16 |
| Figura 6: Caso de uso de un actor con el rol de responsable, padre o tutor | 17 |
| Figura 7: Diagrama de actividad para el rol de persona con autismo | 26 |
| Figura 8: Diagrama de actividad de un usuario con el rol responsable, padre o tutor | 28 |
| Figura 9: Pantalla de inicio de aplicación y de consulta de la tarea actual | 30 |
| Figura 10: Ventanas emergentes: ayuda y confirmación de la salida de la aplicación | 30 |
| Figura 11. Pantallas de configuración y consulta de tareas..... | 31 |
| Figura 12: Pantallas emergentes de selección de día y hora | 31 |
| Figura 13: Pantallas con la lista de tareas predefinidas y su relación con cada franja horaria | 32 |
| Figura 14: Modelo de base de datos | 33 |
| Figura 15: Esqueleto proyecto de “ <i>Agenda Comunicativa</i> ” en Eclipse..... | 36 |
| Figura 16: Actividad “ <i>Agenda</i> ”..... | 36 |
| Figura 17: Intercambio de datos entre actividades..... | 37 |
| Figura 18: Extracto del fichero <i>AndroidManifest.xml</i> de la ‘ <i>Agenda Comunicativa</i> ’. | 38 |
| Figura 19: Componente UI <i>TextView</i> | 39 |
| Figura 20: Componente UI <i>Button</i> | 40 |
| Figura 21: Ejemplo de los componentes <i>TextView</i> y <i>Button</i> | 40 |
| Figura 22: Componente UI <i>ImageButton</i> | 41 |
| Figura 23: Ejemplo componente <i>Imagebutton</i> | 41 |
| Figura 24: Componente UI <i>Spinner</i> | 42 |
| Figura 25: Contenido de “ <i>res/values/arrays</i> ”. | 42 |
| Figura 26: Spinners para la selección de los días de la semana y la hora de realización de una tarea. | 42 |
| Figura 27: Código de un <i>listener</i> que muestra un mensaje en forma de ventana emergente ... | 43 |
| Figura 28: Mensaje en forma de ventana emergente..... | 43 |
| Figura 29: Componente de la interfaz de usuario <i>ListView</i> | 44 |



| | |
|---|----|
| Figura 30: <i>ListView</i> de las tareas disponibles..... | 44 |
| Figura 31: Código que muestra la ayuda en forma de ventana emergente. | 45 |
| Figura 32: Código para pedir confirmación de salida de la aplicación en forma de ventana emergente. | 45 |
| Figura 33: <i>AlertDialog</i> correspondientes al botón <i>Ayuda</i> y al botón <i>Salir</i> respectivamente. ... | 46 |
| Figura 34: Menú de aplicaciones de Android y menú de aplicaciones para personas con necesidades especiales..... | 48 |
| Figura 35: Pantalla de inicio de la <i>Agenda Comunicativa</i> y configuración de tareas..... | 49 |
| Figura 36: Pantalla de selección de día y hora | 50 |
| Figura 37: Pantalla de selección de una de las tareas disponibles | 50 |
| Figura 38: Selección del día a consultar y tareas registradas en un día | 51 |
| Figura 39: Pantalla selección eliminar tarea de un día determinado..... | 52 |
| Figura 40: Pantalla de ayuda y pantalla de confirmación de salida de la aplicación | 52 |
| Figura 41: Pantalla de inicio y consulta de la tarea actual para una persona autista..... | 53 |
| Figura 42: Variables del sistema en Windows | 60 |
| Figura 43: Edición de la variable PATH en Windows..... | 60 |
| Figura 44: Pantalla de Eclipse para añadir el plugin de Android..... | 60 |



1. MOTIVACIÓN

El autismo es un trastorno del desarrollo permanente que impacta en el desarrollo normal del cerebro dentro de las áreas relacionadas con la interacción social, las habilidades comunicativas y la organización de las actividades de la vida diaria.

El *trastorno generalizado del desarrollo (TGD)*, engloba un grupo de discapacidades del desarrollo generalmente detectadas antes de los tres años. Estos trastornos son neurológicos afectando a capacidades del niño como comunicación, imaginación, comprensión del lenguaje, y su relación con el entorno y las personas que le rodean.

Las causas del TGD son variadas: los factores genéticos, infecciones víricas durante los primeros meses de embarazo, complicaciones en el parto, etc. Sin embargo, no existe ninguna hipótesis clara sobre el origen de estos trastornos.

El tratamiento de todas las clases de TGD es un proceso continuo que consiste en la adquisición de habilidades y destrezas. No existe una única terapia que sirva para todos los pacientes. En general, la terapia se ajusta a cada paciente en la cual forman un papel vital tanto el terapeuta como la familia de la persona con TGD.

Los términos autismo y TGD se usan indistintamente tanto en medios de comunicación como en entornos profesionales, aunque hay que recalcar que autismo es un caso particular de los TGDs. Se identifican cinco clases de TGDs:

1. *Síndrome autista*: síndrome caracterizado por el trastorno generalizado del desarrollo que afecta a la comunicación y a las relaciones del niño respecto al entorno que le rodea. En concreto, los síntomas identificativos son problemas en la comunicación verbal y no verbal, en la imaginación, conductas lineales y actividades estereotipadas. Otros síntomas menos usuales son comportamientos violentos, autolesiones o dar vueltas sobre sí mismo.



Este tipo de trastorno se detecta a partir del sexto mes de vida reproduciéndose a lo largo de toda la vida. El origen del mismo es desconocido, por lo que su diagnóstico sólo es posible una vez reproducidas las conductas anteriormente expuestas.

En el autismo existen varios niveles, desde casos en que no existe el habla, comportamientos repetitivos o incluso, como se ha dicho anteriormente, auto dañino o agresivo. También existen casos leves que pueden no detectarse e incluso confundirse con la timidez o la falta de atención. En torno al 25% de casos desarrollan epilepsia cuando se alcanza la edad adulta.

La enfermedad afecta a 4 de cada 10.000 personas, afectando 4 veces más al sexo masculino que al femenino. Este síndrome también se conoce como autismo de *Kanner*.

2. *Síndrome de Asperger*: trastorno de personas que no son capaces de desarrollar la empatía, es decir, no son capaces de recopilar información de los estados emocionales de las personas que le rodean. Por ejemplo, pueden encontrar dificultades para reconocer la ironía o el humor, incluso en casos extremos, no se llega a reconocer gestos faciales como una simple sonrisa.

Este síndrome es un trastorno que suele detectarse más tarde que el trastorno autista ya que no muestra un retraso significativo en el desarrollo del lenguaje. Una persona con este síndrome presenta un nivel de inteligencia normal o superior al normal, en ocasiones se manifiesta en la focalización en un tema de interés, en el cual centran su actividad llegando a ser verdaderos expertos.

3. *Síndrome de Rett*: trastorno congénito que se manifiesta principalmente en niñas al tener un defecto en un gen que está en el cromosoma X. El sexo femenino posee dos cromosomas X por lo que puede sobrevivir cuando uno de ellos está afectado. Por el contrario el sexo masculino sólo posee un cromosoma X, lo que implica que si se encuentra afectado, en la mayoría de las ocasiones el feto no sobrevive.

El desarrollo de estas personas transcurre de forma normal hasta los 12-18 meses, momento en el cual se reproducen pérdidas de habilidades básicas como caminar, a



menudo asociado a un retraso mental de diferente grado dependiendo de caso. Esta enfermedad afecta a 1 de cada 10.000 personas.

4. *Síndrome desintegrativo de la niñez*: trastorno que se manifiesta en una regresión de la conducta general del niño tras 3-4 años de desarrollo normal. En algunos casos, se acorta el diagnóstico a 2 años de edad y en otros se manifiesta en edades hasta los 9 años.

La pérdida de funciones es generalizada tendiendo a estabilizarse con el tiempo, recuperándose aunque de forma muy limitada. Este síndrome presenta similitudes con el autismo. Las diferencias son, que en el síndrome desintegrativo de la niñez el desarrollo físico y cognitivo normal es mayor. La regresión implica la pérdida de otras capacidades además de interacción y relaciones sociales.

5. *Trastorno del desarrollo no especificado*: categoría en la que se engloban los trastornos que cumplen los requisitos de la descripción general de TGD pero no se ajustan a ninguno de los síndromes anteriores.

Como se ha podido ver, dos de las mayores dificultades de las personas autistas son la dificultad de comunicación con el resto de personas de su entorno y la dificultad de organización o planificación de actividades. Con el fin de facilitar estas labores de comunicación y organización, una estrategia consiste en el uso de ayudas visuales para poderse comunicar y establecer el orden en que deberían realizarse tareas cotidianas. Estas ayudas visuales consisten en dibujos simples (uno por concepto) empleando preferiblemente colores y situaciones reales. Las ayudas visuales se emplean en sistemas alternativos y aumentativos de comunicación con distintos objetivos:

- Favorecen la elección de los alumnos entre distintas opciones.
- Señalar los acontecimientos más importantes.
- Trabajar aspectos sociales.
- Dar instrucciones o normas.
- Trabajar las emociones y comprender el concepto del tiempo.

En el libro “*La atención educativa al alumnado con trastorno del espectro autista*” (más información sobre este libro en el apartado de “Referencias bibliográficas” de esta memoria) se puede encontrar más detalles sobre este tipo de ayudas para personas autistas.

Las ayudas visuales pueden realizarse en plantillas físicas dentro de un folio dibujando los conceptos sobre los que la persona autista tiene que trabajar, en concreto para la visualización de actividades a realizar. Es habitual que para ayudar a la persona autista a establecer el orden de actividades diarias tales como lavarse las manos, comer, recoger la mesa, lavarse los dientes, etc., se tenga como apoyo un folio con los dibujos correspondientes a la secuencia de actividades a realizar. Los dibujos muchas veces se realizan a mano o se obtienen de Internet. También existe la posibilidad de realizar este tipo de plantillas con *software* específico como por ejemplo *BoardMaker*. Este *software* es una herramienta para diseñar y adaptar actividades a realizar por personas con necesidades especiales. Contiene más de 4500 símbolos pictográficos en 44 idiomas. En la figura 1, se puede ver un ejemplo de las ayudas visuales que ofrece este programa. Sin embargo, este tipo de SW es bastante caro.

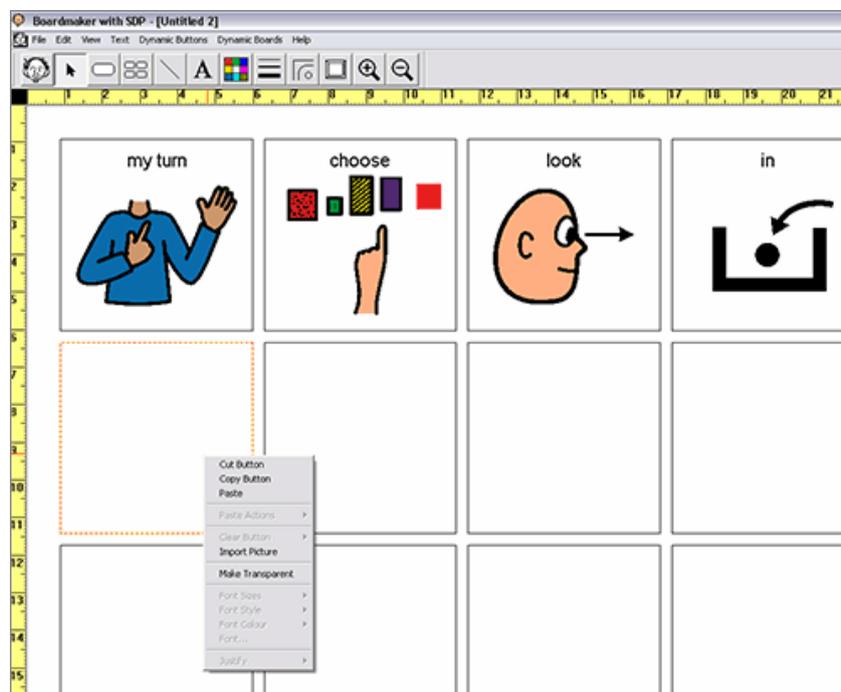


Figura 1: Ejemplo de ayudas visuales en el programa *BoardMaker*

Por otro lado, en los últimos años, gracias a las tecnologías inalámbricas y a la evolución de los dispositivos móviles han surgido muchas aplicaciones destinadas a estos dispositivos que



permiten poder interactuar con ellas en cualquier momento y lugar. En el caso de las personas con autismo sería muy útil disponer de una agenda comunicativa en la que se puedan programar tareas semanales, consultadas posteriormente por las personas que padecen autismo o TGD. El objetivo de que se utilice un dispositivo móvil, es que la persona autista siempre pueda llevar consigo su agenda comunicativa y pueda consultarla independientemente de donde esté. De esta forma, se sustituyen los voluminosos y poco atractivos archivadores de tareas, en los cuales se incluyen las actividades de las personas con autismo clasificadas por días.

A continuación, se detalla la estructura del presente documento:

- La sección 2 describe brevemente la problemática a resolver, los requisitos de la aplicación teniendo en cuenta las capacidades de estos usuarios, el estudio de las alternativas realizado y la metodología empleada.
- La sección 3 muestra el detalle de la descripción informática. En concreto, la identificación y descripción de casos de uso, los diagramas de actividad, el diseño de base de datos, el diseño de bocetos preliminares sobre la interfaz gráfica de la implementación, la propia implementación, y por último, una representación de diferentes escenarios posibles.
- Finalmente se exponen las conclusiones y trabajos futuros.

También hay que indicar que existe un anexo, en el cual se muestra el proceso de configuración de entorno de desarrollo.



**Agenda comunicativa para personas con TGD o
autismo mediante dispositivos móviles**

Tutora: Estefanía Martín
Alumno: Miguel A. Santiago



2. OBJETIVOS

2.1 Descripción del problema

Tal y como se ha indicado previamente en la sección de motivación, los dispositivos móviles son una herramienta útil para el día a día de las personas con autismo, formando parte incluso en las distintas terapias que se aplican a estas personas. En concreto en el caso del autismo, uno de los puntos importantes es la planificación de tareas a realizar por el paciente, usando para ello hojas de papel o archivadores en donde se clasifican las actividades de cada día. Por este motivo, la “*Agenda Comunicativa*” nace con el propósito de digitalizar este proceso haciendo uso para ello de dispositivos móviles. Esto facilitará esta labor a las personas a cargo del paciente y hará más atractivo el seguimiento de las tareas al propio paciente.

2.2 Requisitos de usuario

La aplicación “*Agenda Comunicativa*” nace con el objetivo de facilitar tanto la comunicación como la gestión del día a día de personas que padecen TGD. Por ello, tanto los requisitos funcionales como de contenido giran en torno a esta tipología de usuarios, en concreto, la aplicación, y por extensión la interfaz gráfica, está dividida en dos roles: “rol de persona con TGD” y “rol de padre, responsable o tutor”. El primero se corresponde con pantallas exentas de texto basándose toda la interacción con el usuario en forma de imágenes. La funcionalidad de estas pantallas es la mínima posible. Por ejemplo, si la acción a realizar es consultar una tarea, el usuario simplemente tendrá que pulsar sobre la pantalla táctil para obtener la información solicitada. En cuanto al segundo rol de usuarios, el interfaz gráfico no será tan restrictivo como el anterior ya que contendrá tanto imágenes como texto para poder, de esta forma, configurar la agenda de cara a la posterior consulta del usuario final con TGD.

Dentro de las características deseables de una agenda, la más importante sin duda es la movilidad de la misma, es decir, que se pueda consultar en el momento deseado. La movilidad es fundamental, pero no tendrá razón de ser si no está asociada a otra serie de características secundarias como puede ser el tamaño, la capacidad de gestión de procesos y de memoria, entre otros. La conjunción de todos estos aspectos hace que los dispositivos



móviles sea el soporte *hardware* ideal para el desarrollo de una agenda, para el caso concreto de “*Agenda Comunicativa*”, una agenda para personas que padecen TGD.

Por último, indicar que los datos almacenados en la aplicación, se reducen al almacenamiento de tareas que realizan las personas con TGD y la asociación de éstas con sus respectivos horarios. Las primeras, el listado de todas las tareas, se encuentran precargadas en la aplicación. En cambio, la segunda tipología de datos almacenados, son las tareas y su día-hora asociado que se registran conforme el rol padre, responsable o tutor las configure. De esta forma, en caso de *reset* del dispositivo móvil se podrá instalar de nuevo la aplicación con las tareas precargadas, perdiendo, por el contrario, la asociación de las mismas al día y hora previamente configuradas.

2.3 Estudio de alternativas

A continuación se detallarán tanto las alternativas barajadas como la motivación de la elección de *Android* como plataforma de desarrollo de la aplicación “*Agenda Comunicativa*”. Para hacerse una composición de lugar, se detallan brevemente las principales características, ventajas e inconvenientes de cada una de las alternativas. Se hará hincapié en si son *opensource* y si facilitan el desarrollo de aplicaciones de terceros ya que ambas son características fundamentales para el desarrollo de este Proyecto Fin de Carrera.

Algunos de los sistemas operativos existentes para dispositivos móviles son *iPhone OS*, *Microsoft Windows Mobile*, *Symbian OS*, *BlackBerry OS*, o *Android*, entre otros. Las características de cada una de las alternativas se exponen a continuación:

- **iPhone OS:** sistema operativo de los dispositivos móviles de Apple usando como base el kernel de *MACOS X*¹, con una arquitectura *ARM* mediante la cual se permite el consumo energético reducido, cualidad fundamental para este tipo de dispositivos. Además posee una velocidad muy alta comparada con sus competidores en el interfaz de usuario. Por el contrario, cuenta con desventajas como que no es *opensource* aunque su SDK se puede descargar gratis. Para liberar el *software* es necesario registrarse en el programa de desarrollo de *iPhone* siendo esta última acción de pago.

¹ MACOS X: Sistema operativo de Apple Inc.



En cuanto a desarrollos por parte de terceros, *Apple* dispone de un lugar *online* donde distribuirlos. Sin embargo, en el caso de un proyecto académico, no es una ventaja ya que conlleva costes previos a la publicación/distribución.

- **Microsoft Windows Mobile:** sistema operativo creado por la compañía *Microsoft* con un interfaz de usuario similar a *Windows* de los ordenadores personales. Cuenta con aplicaciones similares como *Office Mobile* u *Outlook Mobile*, siendo esto una ventaja de cara a la transición de los usuarios de *Windows*. En cuanto al desarrollo de terceras personas es necesario adquirir herramientas propietarias como *Microsoft Visual Studio*.
- **Symbian OS:** sistema operativo fruto de la unión de varias compañías de telefonía móvil, siendo la mayor accionista *Nokia*. En cuanto a las ventajas, ofrece distintas *run-times* para el desarrollo de aplicaciones por terceros en diferentes lenguajes de programación como *Python*, *Ruby*, etc., pudiendo descargar la *SDK* en la que se incluyen las librerías necesarias, documentación, ejemplos, emulador de móvil e incluso un *IDE* gratuito. Por el contrario, *Symbian* define una serie de familias de interfaces de usuario dividiendo de esta forma, distintos grupos de móviles con sus sistemas operativos correspondientes. Este hecho implica adaptar cada aplicación a cada una de estas familias.
- **BlackBerry OS:** sistema operativo creado por la compañía *Research In Motion (RIM)* siendo una plataforma *J2ME*, *suite* reducida de Java orientado a dispositivos móviles. BlackBerry ofrece de forma gratuita el entorno de desarrollo, teniendo acceso tanto a las APIs de *J2ME* como a las propietarias de *BlackBerry*. Este entorno de desarrollo incluye un simulador de móvil, aunque hay que señalar que si la aplicación hace uso de determinadas funciones o librerías debe ser firmada previo pago de la misma.
- **Android:** es un sistema operativo basado en núcleo *Linux*. Esto significa que se alimenta de su gestión de procesos y de memoria, así como la interacción entre el *hardware* y el *software* del mismo. Aparte de la capa del *kernel* basada en *Linux*, existen otras capas de abstracción del sistema operativo, librerías nativas como librerías gráficas, base de datos (p.e. *SQLite*) y el *Android runtime* encargado de recibir y gestionar las peticiones de las aplicaciones (ver figura 2). Por encima de esta

capa, se encuentran las aplicaciones por defecto como puede ser gestor de música, mensajes y las aplicaciones creadas por terceros como es el caso de “*Agenda Comunicativa*”.

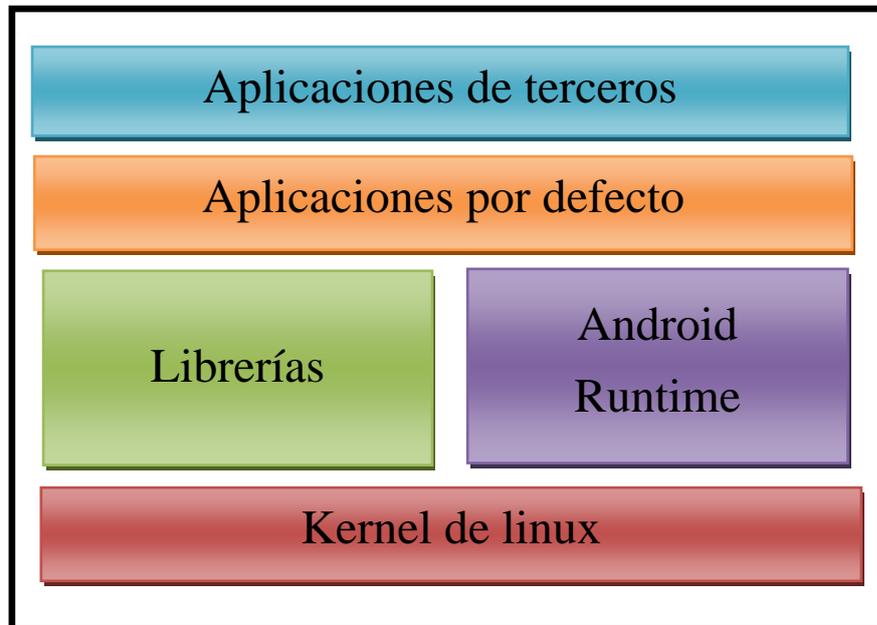


Figura 2: Estructura de Android.

En su inicio, *Android* se concibió para dispositivos móviles pero su uso se ha extendido a otros ámbitos como ordenadores personales, televisores, incluso electrodomésticos como lavadoras o microondas.

Android es una plataforma de código abierto bajo licencia de *Apache*² creada por *Android Inc.* Esta compañía fue adquirida posteriormente por Google, la cual ha sido la que ha publicado la mayor parte del código.

Google no viaja sólo en este proyecto, colaboran entre otros, compañías como Motorola, Samsung, Texas Instruments, T-Mobile o Telefónica. Como se puede observar, son empresas punteras a nivel mundial en el sector de las telecomunicaciones incluyendo proveedores de servicios telefónicos y fabricantes de dispositivos móviles. Esta amplia colaboración junto

² Licencia de Apache: Licencia de software libre creada por Apache Software Foundation.



con la característica de ser código abierto hace que *Android* atraiga cada vez más usuarios, desde que viera la luz en noviembre del 2007.

La plataforma Android provee a los desarrolladores un SDK (*Software Development Kit*) que contiene las herramientas, códigos de ejemplo, documentación, tutoriales, librerías, emulador de teléfono móvil y la API³ necesaria para desarrollar las aplicaciones. A continuación, se enumeran las características más significativas de esta plataforma:

- Máquina virtual *Dalvik*⁴ para dispositivos móviles.
- Diferentes librerías para el acceso a funciones, acceso a datos del terminal, manipulación de textos o elementos predefinidos de la interfaz de usuario.
- Navegador Web integrado.
- Gráficos en 2D y 3D bajo el estándar *OpenGL*.
- Soporte para formatos de imagen, audio y video con la capacidad de soporte streaming⁵
- Acceso a datos de localización como puede ser GPS u otro servicio proveedor de geolocalización.
- Acceso y manipulación de bases de datos *SQLite*.
- Compatibilidad con *Wifi*, *Bluetooth* y 3G.
- Soporte *XMPP* (*extensible Messaging and Presence Protocol*). Este protocolo se utiliza para mensajería instantánea pero *Android* lo explota para la comunicación entre aplicaciones.
- Soporte *MapView* (navegador de mapas geográficos).

³ API: Interfaz de programación de aplicaciones, es decir, grupo de rutinas y métodos que se pueden realizar con una determinada aplicación o biblioteca.

⁴ Dalvik: Máquina virtual creada “ad hoc” para dispositivos Android optimizada para requerir poca memoria.

⁵ Streaming: Transmitir de forma remota video o audio a través de la red en tiempo real a la vez que se encuentra descargando.



Todos estos componentes de *Android* son multiplataforma, es decir, funcionan en distintos sistemas operativos como Windows, Linux o Mac OS.

También se encuentra disponible la NDK (*Native Development Kit*) complemento de la anteriormente mencionada SDK. Mediante esta serie de complementos, los desarrolladores pueden implementar aplicaciones utilizando directamente código nativo. Esto último implica la posibilidad de desarrollar aplicaciones en otros lenguajes distintos a Java como puede ser C. La NDK se encuentra disponible para Windows, Linux y Mac OS, incluyendo para todos ellos librerías, ejemplos y documentación.

Existen diferentes versiones de Android, en concreto la aplicación “*Agenda comunicativa*” se ha desarrollado en la versión Android 2.0. Se podría haber usado la versión estable anterior Android 1.6, ya que la versión 2.0 se basa en mejoras en mensajería instantánea, navegador Web, gestión de mensajería, gestión de contactos, optimización de la gestión de la cámara de fotos y arreglo de *bugs* en el *Bluetooth* que no son usadas en esta aplicación.

En resumen, *Android* ofrece una plataforma de desarrollo de código libre, un emulador y una plataforma de herramientas para el desarrollo de aplicaciones orientadas a dispositivos móviles, respaldado por una gran comunidad tanto de usuarios como desarrolladores.

Una vez se conocen las características del sistema operativo *Android* y los trastornos específicos de los pacientes con TGD o autismo, se puede afirmar que esta tecnología puede ayudar a mejorar la calidad de vida de estos niños. ¿Cómo? En concreto, el proyecto final de carrera “*Agenda comunicativa*” puede ayudar a la planificación de actividades para la vida diaria, siendo éste uno de los puntos vitales de la mayoría de terapias aplicadas en este tipo de enfermedades.

2.4 Metodología empleada

En el proceso de desarrollo de aplicaciones *software* está extendido el uso de **patrones de diseño**, es decir, cada definición de patrón es una solución a un conjunto de problemas con características comunes.

En este sentido, *Android* hace uso del patrón “*Modelo Vista Controlador*” por lo que por extensión, la aplicación *Agenda Comunicativa* hace uso del mismo. A continuación, se enumeran las características principales de este patrón junto con su implementación concreta en esta aplicación.

El patrón “*Modelo Vista Controlador (MVC)*” permite separar la lógica de control, la lógica de negocio y la lógica de presentación (ver figura 3). En concreto:

- El modelo: incluye la lógica de negocio y los datos de la aplicación.
- El controlador: es el encargado de atender peticiones y decisiones.
- La vista se encarga de la presentación de los datos.

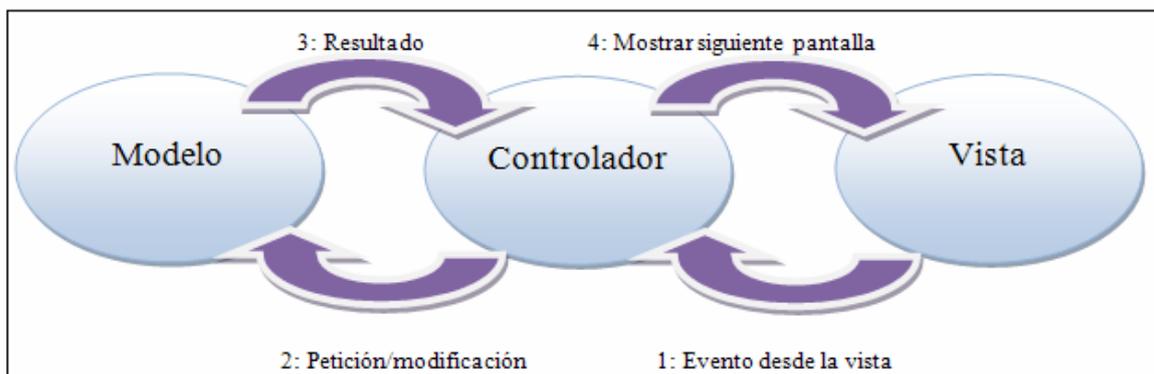


Figura 3: Esquema MVC.

Con esta estructura, se permite realizar código de calidad, perdiendo del temor al “folio en blanco” del inicio de cualquier desarrollo. Además, los datos y la lógica de negocio se encuentran desacoplados de la lógica de presentación, es decir, hay una clara separación entre los componentes de una aplicación, lo cual facilita su mantenimiento y/o actualización.

En cuanto al caso de concreto de *Android*, y por extensión en la “*Agenda Comunicativa*”, se refleja en un MVC donde la vista se define mediante contenedores *xml* haciendo uso de

componentes predefinidos o definidos por el propio programador. Sin embargo, la lógica de negocio se define mediante la implementación de clases *Java* que heredan de cada una de las clases del propio API de *Android*. Estas clases (por ejemplo, las clases *Activity* y *ListActivity*) definen las funcionalidades de las actividades. Por último, la capa del modelo de datos hace uso del sistema de gestión de datos *SQLite*, una base de datos relacional cuya principal característica es que en un sólo fichero se guardan los datos, las tablas y los índices. Esto hace un sistema tremendamente liviano ideal para dispositivos móviles.

En la figura 4, se refleja cómo se ha estructurado la “Agenda Comunicativa” siguiendo el patrón “*Modelo Vista Controlador*”. Tal y como se puede observar, existe una clara separación entre los distintos componentes.

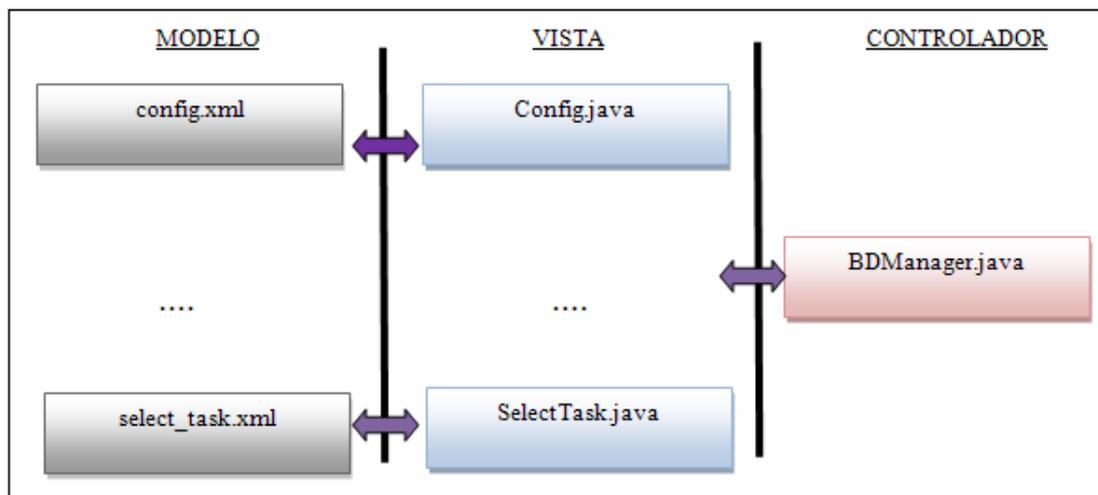


Figura 4: Esquema MVC de la aplicación “Agenda Comunicativa”



3. DESCRIPCIÓN INFORMÁTICA

Con el objetivo de implementar la aplicación “*Agenda Comunicativa*” se realizarán una serie de pasos hasta conseguir la implementación de la misma:

1. Identificación de los diferentes casos de uso de la aplicación y los diagramas de actividad.
2. A partir de los casos de uso y los diagramas de actividad del punto anterior, diseñar un boceto de las pantallas de la “*Agenda Comunicativa*”.
3. Diseño de la base de datos.
4. Implementación de la aplicación.

Tanto los escenarios de interacción que describen los casos de uso como los diagramas de actividad y sus bocetos de pantallas correspondientes, se han realizado de forma paralela para de esta forma refinar sucesivamente el producto hasta llegar a la versión final de la aplicación.

3.1 Casos de uso

En la aplicación de la *Agenda Comunicativa* se diferenciarán dos diferentes tipos de usuarios que interactuarán con la aplicación: el usuario con el rol de tutor, padre o responsable y el usuario con el rol de niño con autismo o TGD.

En cuanto al primer rol, el actor tiene acceso a toda la funcionalidad de la aplicación desde visualización de tareas hasta la configuración de las mismas. En cambio, la persona autista sólo podrá consultar las tareas previamente configuradas por el rol tutor, padre o responsable.

3.1.1 Diagramas de casos de uso

En este apartado se han diferenciado dos diagramas de casos de uso correspondientes a cada uno de los actores. En el primer diagrama se muestran los casos de uso para un actor con rol

de persona autista. Una persona con autismo sólo podrá realizar la acción de consultar la tarea en curso y reproducir su correspondiente ayuda en forma de audio.

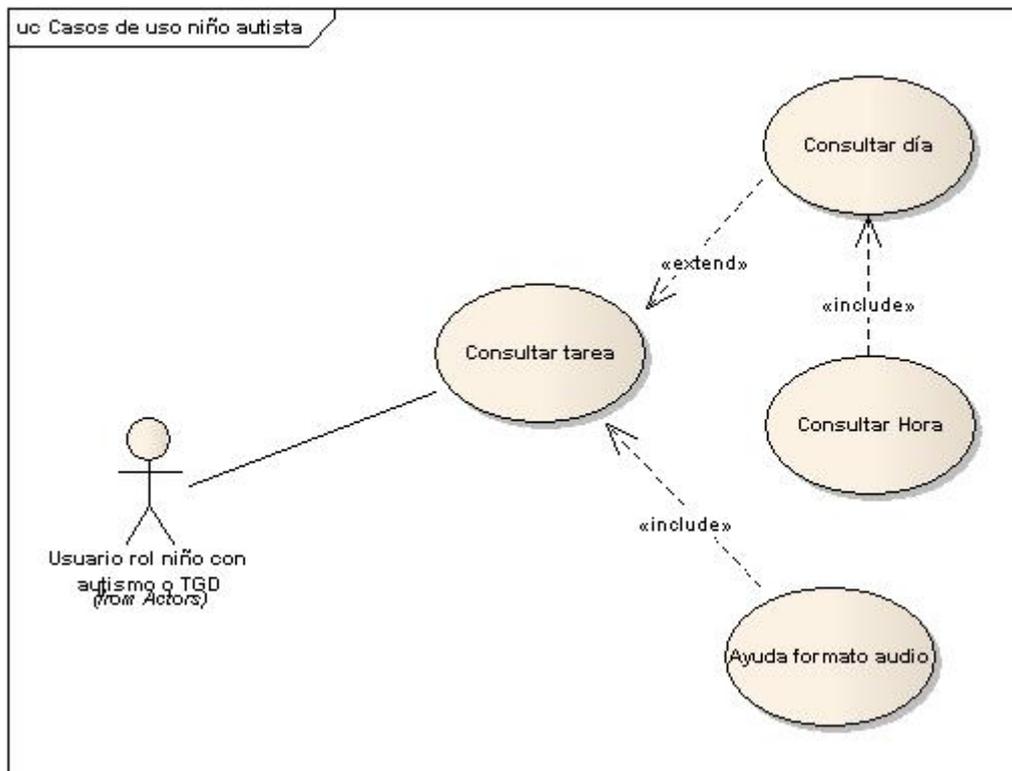


Figura 5: Casos de uso de un usuario autista

El segundo diagrama describe los casos de uso del usuario con rol tutor, padre o responsable (véase figura 6). Como se puede observar, el responsable de la persona con autismo puede realizar todas las acciones posibles de la aplicación, es decir, puede acceder a la consulta de tareas y a la planificación de las mismas.

Dentro de este diagrama cabe destacar el caso de uso “Mostrar menú aplicaciones”, el cual muestra un menú de inicio con aplicaciones dirigidas a personas con necesidades especiales. En este menú se incluye tanto la aplicación correspondiente a este documento “*Agenda comunicativa para personas con TGD o autismo mediante dispositivos móviles*” como la aplicación “*Dispositivo de ayuda a la comunicación, para personas con discapacidad auditiva, sin intérprete, u oyentes con desconocimiento del idioma del receptor*” realizada por Jebel Gil. Esta última aplicación facilita la comunicación entre el emisor y receptor en

escenarios especiales como por ejemplo, cuando alguno de los interlocutores tiene problemas de audición o desconoce el idioma.

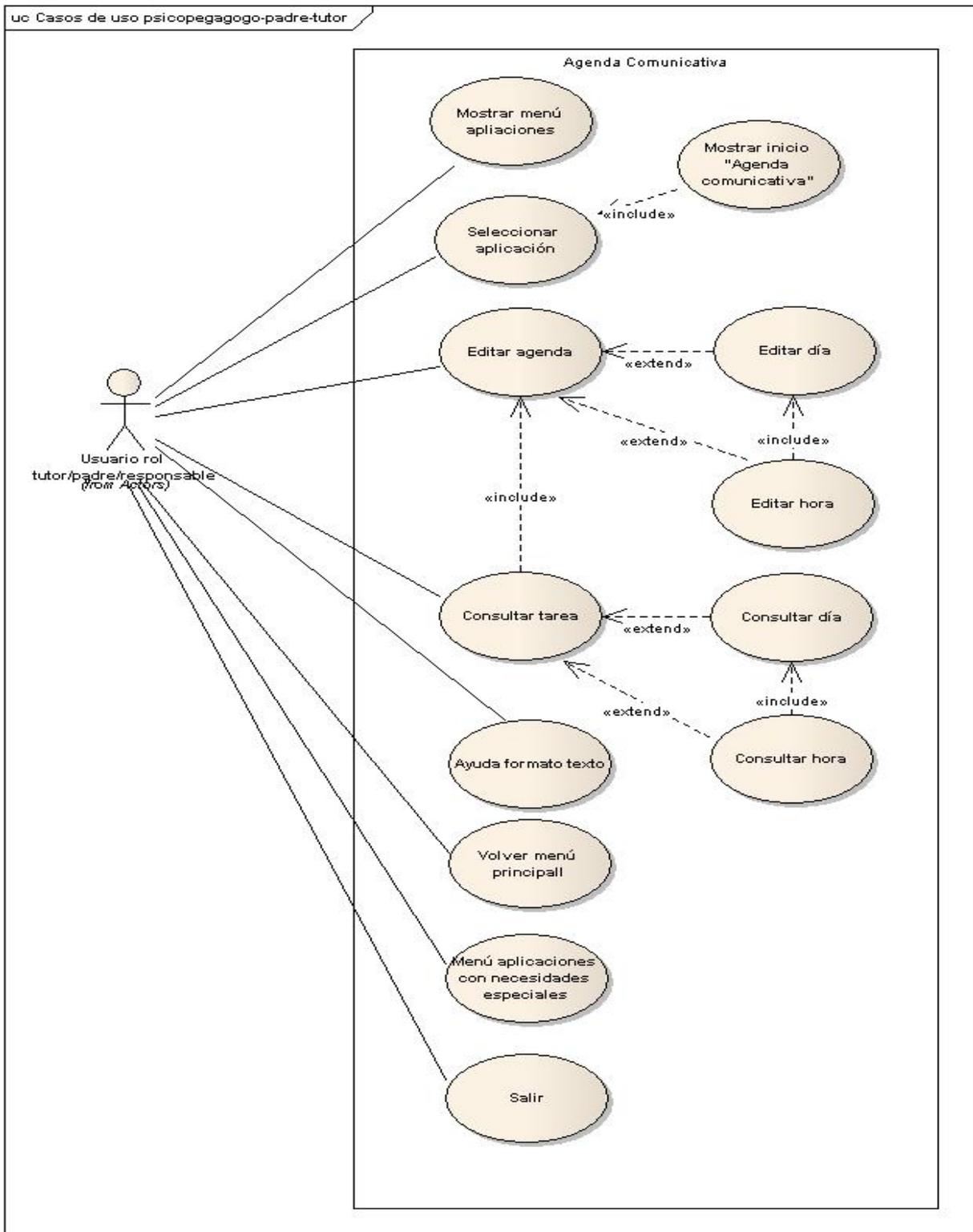


Figura 6: Caso de uso de un actor con el rol de responsable, padre o tutor



3.1.2 Descripción casos de uso

En este subapartado, se muestra una descripción de cada caso de uso incluyendo los actores implicados, las precondiciones necesarias, flujo normal, flujo alternativo así como las postcondiciones a cumplir de cada uno de ellos.

Nombre: Mostrar Menú Aplicaciones

Descripción: Permite mostrar las aplicaciones relativas a personas con necesidades especiales

Actores: Usuario rol responsable-padre-tutor

Actores: Usuario rol responsable-padre-tutor

Precondiciones: El actor debe encontrarse en el menú principal de las aplicaciones del móvil.

Flujo normal:

1. El actor pulsa sobre el icono “Aplicaciones para personas con necesidades especiales”
2. El sistema muestra el menú con las dos aplicaciones:
 - Agenda comunicativa
 - Dispositivo de ayuda a la comunicación

Flujo alternativo:

No aplica

Postcondiciones: El sistema muestra los dos botones, correspondientes a cada una de las aplicaciones desarrolladas “*Agenda comunicativa*” y “*Dispositivo de ayuda a la comunicación*”, así como otros dos botones, “Ayuda” y “Salir”



Nombre: Seleccionar Aplicación.

Descripción: Permite ejecutar la aplicación relativa a personas con necesidades especiales seleccionada.

Actores: Usuario rol responsable-padre-tutor.

Precondiciones: El actor debe encontrarse en el Menú Aplicaciones, al cual se ha accedido mediante el caso de uso “Mostrar Menú Aplicaciones”.

Flujo normal:

1. El actor pulsa sobre el icono de la aplicación que quiere ejecutar.
2. El sistema lanza esta aplicación.

Flujo alternativo:

- El actor pulsa sobre el botón “Ayuda”, cuyo comportamiento será descrito en el caso de uso del mismo nombre.
- El actor pulsa sobre el botón “Salir”, cuyo comportamiento será descrito en el caso de uso del mismo nombre.

Postcondiciones: El sistema lanza la aplicación seleccionada por el actor.



Nombre: Mostrar inicio “Agenda comunicativa”.

Descripción: Muestra la pantalla inicio de la aplicación Agenda comunicativa.

Actores: Usuario del dispositivo con S.O. Android.

Precondiciones: El usuario se encuentra en la pantalla de menú definido en el caso de uso “Seleccionar Aplicación”.

Flujo normal: Este caso de uso replica la funcionalidad del caso de uso común “Seleccionar aplicación” pero recalca la diferencia entre ambos PFC.

Una vez seleccionada la aplicación “*Agenda Comunicativa*” mediante el caso de uso explicado anteriormente “Seleccionar aplicación”, se muestra la pantalla de inicio de esta aplicación.

Flujo alternativo:

- El actor pulsa sobre el botón “Ayuda”, cuyo comportamiento será descrito en el caso de uso del mismo nombre.
- El actor pulsa sobre el botón “Consulta agenda”, cuyo comportamiento será descrito en el caso de uso del mismo nombre.

Postcondiciones: El sistema muestra una pantalla con un botón central en forma de imagen que ocupa la mayor parte de la pantalla, la cual una vez pulsada sobre ella muestra la tarea a realizar en ese momento. Esto es así para que el niño con autismo o TGD centre su atención y no se distraiga con elementos secundarios.



Nombre: Editar agenda (incluye los casos de uso “Editar día” y “Editar hora”).

Descripción: Acción editar agenda.

Actores: Usuario rol responsable-padre-tutor.

Precondiciones: El actor debe seleccionar previamente el día a editar (caso de uso “Consultar día”).

Flujo normal:

1. El actor selecciona el día en el desplegable en el que se muestran los días.
2. El actor selecciona la hora en el desplegable en el que se muestran las horas.
3. El sistema muestra la pantalla de edición de tarea.
4. El actor selecciona la tarea a editar.
5. El sistema registra la tarea en la base de datos.

Flujo alternativo:

- El actor pulsa sobre el botón “Salir”, cuyo comportamiento será descrito en el caso de uso del mismo nombre.
- El actor pulsa sobre el botón “Ayuda”, cuyo comportamiento será descrito en el caso de uso del mismo nombre.

Postcondiciones: El sistema registra la tarea seleccionada acorde con la hora y el día escogido en los puntos 1 y 2 del apartado “flujo normal”.



Nombre: Consultar agenda (incluye los casos de uso “Consulta día” y “Consulta hora”).

Descripción: Acción consultar la tarea en curso.

Actores: Usuario rol niño con autismo o TGD.

Precondiciones: El actor con rol responsable-padre-tutor debe seleccionar previamente la aplicación “*Agenda comunicativa*”.

Flujo normal:

1. El actor rol niño con autismo o TGD pulsa sobre la imagen descrita en el caso de uso “Mostrar inicio Agenda comunicativa”.
2. El sistema muestra la pantalla con la tarea que se corresponde con la tarea que se ha almacenado para el día y hora del momento de la consulta.

Flujo alternativo:

- El actor rol responsable-padre-tutor podrá volver al menú principal, cuyo comportamiento será descrito en el caso de uso del mismo nombre.
- El actor rol responsable-padre-tutor pulsa sobre el botón “Ayuda”, cuyo comportamiento será descrito en el caso de uso del mismo nombre. Nótese que este botón se sitúa en la parte inferior izquierda, siendo este de tamaño reducido.

Postcondiciones: El sistema muestra la pantalla con la tarea que se corresponde con la tarea que se ha almacenado (siguiendo el caso de uso “Editar agenda”) para el día y hora del momento de la consulta de la misma.



Nombre: Volver Menú Principal.

Descripción: Permite regresar al Menú Principal de la aplicación “Agenda comunicativa”

Actores: Usuario rol responsable-padre-tutor.

Precondiciones: El actor debe encontrarse en cualquier pantalla donde aparezca el botón “Volver Menú Principal”.

Flujo normal:

1. El actor pulsa sobre el botón “Volver Menú Principal”.
2. El sistema regresa al Menú Principal de la Agenda comunicativa.

Postcondiciones: El dispositivo muestra el menú principal de la aplicación.

Nombre: Menú Aplicaciones Personas con necesidades especiales.

Descripción: Permite regresar al menú donde seleccionar aplicaciones relativas a personas con necesidades especiales.

Actores: Usuario rol responsable-padre-tutor.

Precondiciones:

El actor debe encontrarse en el menú principal de la aplicación Agenda comunicativa.

Flujo normal:

1. El actor pulsa sobre el icono “Volver Menú Aplicaciones Personas con necesidades Especiales”.
2. El sistema regresa al menú con las dos aplicaciones:
 - Agenda comunicativa.
 - Dispositivo de ayuda a la comunicación.

Postcondiciones: El sistema muestra los dos botones, correspondientes a cada una de las aplicaciones desarrolladas (Agenda comunicativa), así como otros dos botones, “Ayuda” y “Salir”.



Nombre: Ayuda formato texto.

Descripción: Permite mostrar una descripción de dónde se encuentra el actor, qué opciones tiene y qué significa cada una de estas opciones.

Actores: Usuario rol responsable-padre-tutor.

Precondiciones: El actor debe encontrarse en cualquier pantalla descrita en este documento, a excepción del menú principal de aplicaciones del móvil.

Flujo normal:

1. El actor pulsa sobre el botón “Ayuda”.
2. El sistema muestra en pantalla una descripción en formato de texto de dónde se encuentra el actor y qué opciones tiene, indicando qué significa cada una de ellas.

Postcondiciones: El sistema muestra en pantalla una descripción de dónde se encuentra el actor y qué opciones tiene, indicando qué significa cada una de ellas.

Nombre: Ayuda formato audio.

Descripción: Pequeña descripción en formato audio de la tarea asociada.

Actores: Usuario rol niño con autismo o TGD.

Precondiciones: El actor debe encontrarse en la pantalla descrita en el caso de uso “Mostrar inicio Agenda comunicativa”.

Flujo normal:

1. El actor pulsa sobre la imagen que describe la tarea
2. El sistema reproduce el audio que se ha grabado para la descripción de la tarea en cuestión.

Postcondiciones: El sistema reproduce el audio de descripción de la tarea. La aplicación sigue en la misma pantalla una vez finalizada la reproducción, es decir, se sigue mostrando la tarea en curso.



Nombre: Salir.

Descripción: Permite regresar al menú principal de aplicaciones del móvil.

Actores: Usuario rol responsable-padre-tutor.

Precondiciones: El actor debe encontrarse en cualquier pantalla descrita en este documento, a excepción del menú principal de aplicaciones del móvil.

Flujo normal:

1. El actor pulsa sobre el botón “Salir”.
2. El sistema pide la confirmación de salir de la aplicación en forma de ventana emergente.
3. El sistema regresa al menú principal de aplicaciones del móvil.

Postcondiciones: El dispositivo muestra el menú principal de aplicaciones del móvil.

3.2 Diagrama de Actividad

Los diagramas de actividad sirven para modelar el comportamiento de una operación en términos de acciones. De esta manera, se pueden ver las acciones que el usuario final podrá realizar con la aplicación. En concreto para la aplicación “*Agenda Comunicativa*” se han desarrollado dos diagramas de actividad, uno para el flujo de trabajo con el rol de persona con TGD o autismo; y otro para el rol tutor, padre o responsable.

3.2.1 Diagrama de actividad para un usuario con el rol persona con autismo

Como se puede observar en la figura 7, el punto de partida del diagrama de actividad del usuario con el rol persona con autismo o TGD, se sitúa en la pantalla de inicio de la aplicación “*Agenda Comunicativa*”. Esta decisión se basa en a las dificultades de este tipo de usuario. Por ello, se ha decidido que la persona encargada de su cuidado inicie la aplicación de la agenda para que la persona con el rol de persona con autismo o TGD pueda consultar la actividad actual.

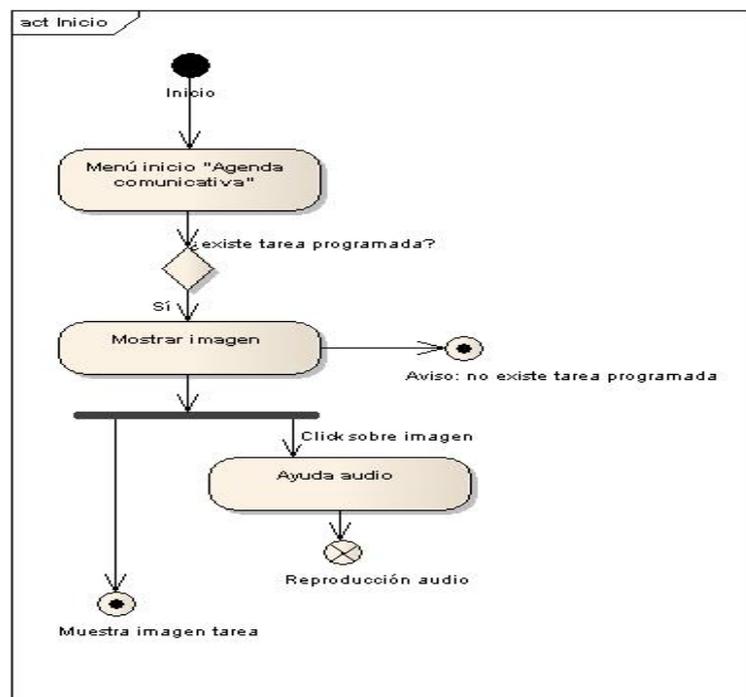


Figura 7: Diagrama de actividad para el rol de persona con autismo



Por otro lado, destacar que existen dos finales de actividad que se corresponde con: i) la visualización en la aplicación de la imagen de la tarea programa y con ii) la visualización de un mensaje de aviso que indica que no existen tareas programadas. Alternativamente a estos dos finales de actividad, existe un final de flujo que se corresponde con la reproducción de la ayuda en formato de audio de la tarea programada para facilitar la comprensión de la tarea a realizar al niño autista.

3.2.2 Diagrama de actividad para un usuario con rol padre, responsable o tutor

En el diagrama de la figura 8 se muestran las actividades que podrá realizar un usuario con el rol de padre, responsable o tutor. Se muestra tanto el flujo desde el punto inicial hasta los diferentes finales de actividad pasando por los finales de flujo. En concreto, tenemos cuatro posibles finales de actividad:

- mostrar la imagen de la tarea programada,
- mostrar un aviso para indicar que no existen tareas programadas,
- mostrar una lista de tareas registradas para cada día y por último,
- registrar una determinada tarea en la base de datos.

En cuanto a los finales de flujo, existen tres: mostrar la ayuda en formato texto, mostrar la ayuda en formato de audio y salir de la aplicación.

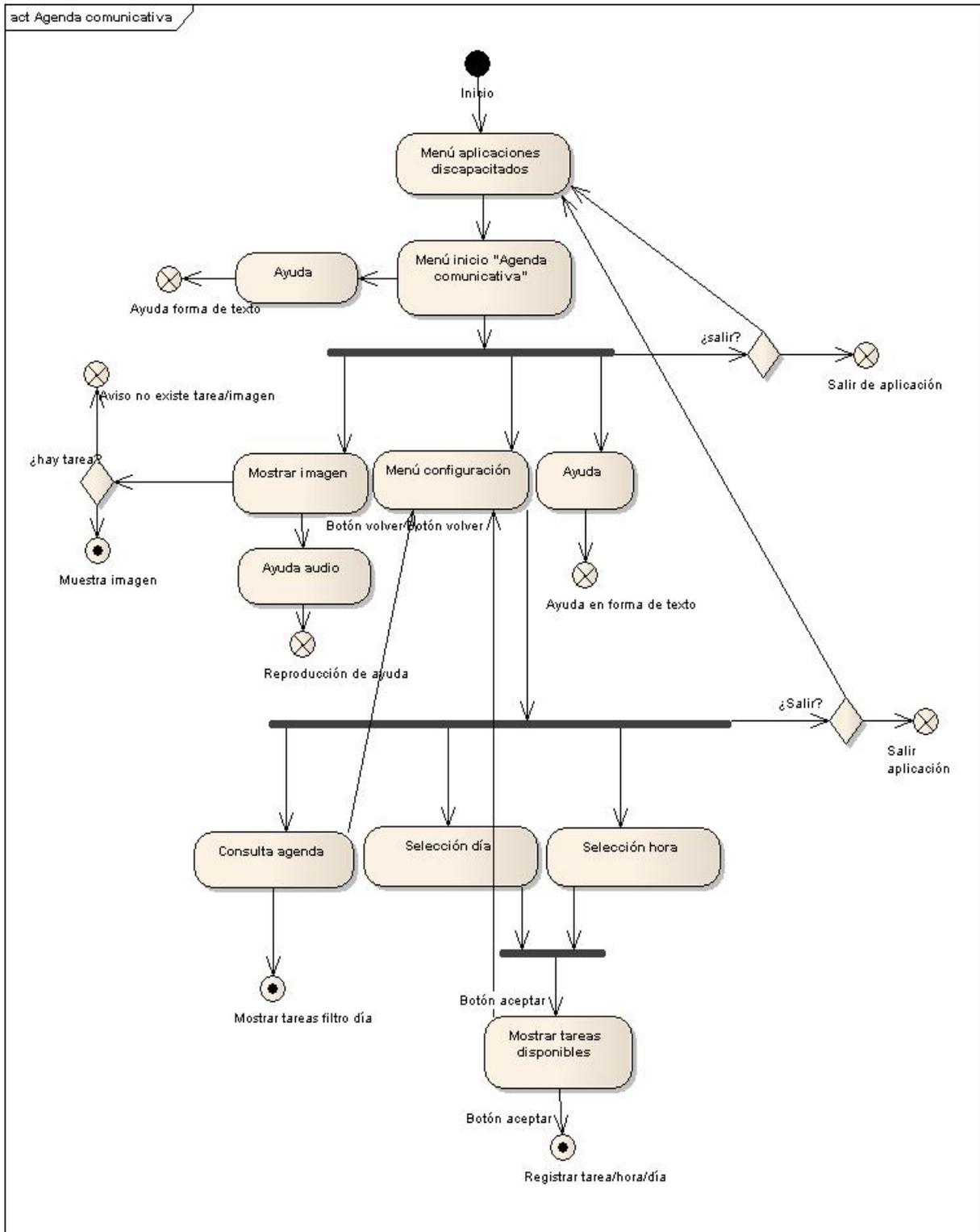


Figura 8: Diagrama de actividad de un usuario con el rol responsable, padre o tutor

3.3 Diseño de bocetos de las pantallas de la aplicación

A la vez que se estudiaron los casos de uso de la aplicación y los diagramas de actividad, se realizaron bocetos de las posibles pantallas de la aplicación con el objetivo de definir los requisitos de la interfaz gráfica e irlos refinando.

Hay que tener en cuenta que no es una representación fiel cien por cien del producto final, ya que se concibieron como guía de la posible interfaz de usuario. Estos bocetos se realizaron con la aplicación “*Balsamiq Mockups*” mediante la cual es posible realizar bosquejos de interfaces gráficas de aplicaciones informáticas de forma fácil y rápida.

En este apartado no se muestran pantallas repetidas en cuanto al diseño de la interfaz de usuario como puede ser las pantallas que muestran la ayuda o la confirmación de salida de la aplicación.

Las figura 9 muestran las posibles pantallas en las que el usuario con rol persona con TGD consulta la tarea a realizar. La versión final de la aplicación además muestra en la parte superior las tareas inmediatamente anteriores a la actual. Éstas serán más pequeñas que la imagen de la tarea en curso para no perder el foco de esta última. En esta misma línea, en la versión final de la aplicación, tanto los botones de salida de la aplicación como de la configuración de la misma se reducen al máximo en la parte inferior de la pantalla para no distraer a la persona autista.

La figura 10 se corresponde con dos ventanas emergentes de la aplicación. En concreto, la ventana que muestra la ayuda en forma de texto (parte izquierda de la figura) y la ventana que pide la confirmación o cancelación de salida de la aplicación respectivamente (parte derecha de la figura).



Figura 9: Pantalla de inicio de aplicación y de consulta de la tarea actual

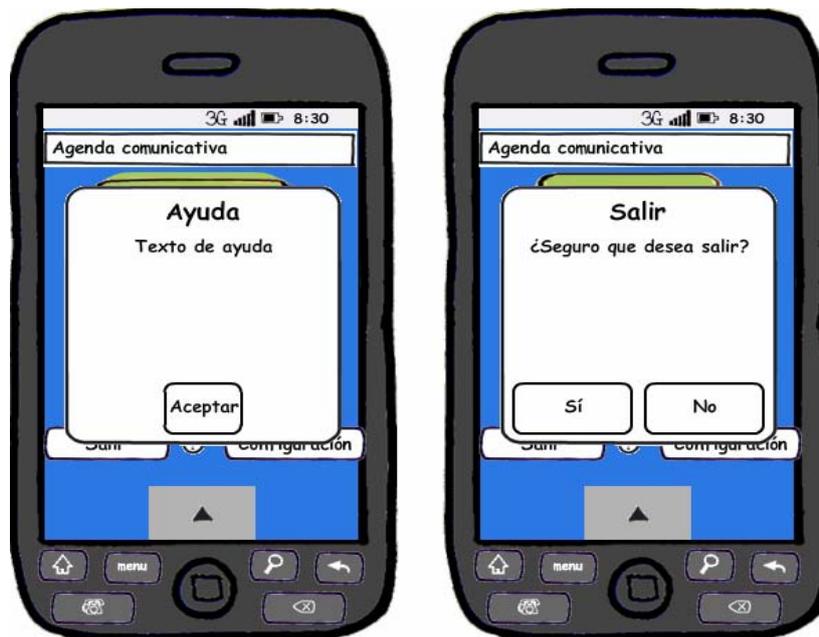


Figura 10: Ventanas emergentes: ayuda y confirmación de la salida de la aplicación

La figura 11 muestra las pantallas de configuración del día y hora de las tareas (ver parte izquierda de la figura) y la pantalla de consulta de tareas previamente registradas informando el día (ver parte derecha de la figura).



Figura 11. Pantallas de configuración y consulta de tareas

La figura 12 muestra los mensajes temporales en la pantalla de la selección previa de uno de los elementos. En este caso concreto, se exhibe un mensaje en forma de ventana emergente la selección del día y la hora.



Figura 12: Pantallas emergentes de selección de día y hora

Por último, se puede observar en la figura 13 las pantallas que listarán las tareas prefijadas en la base de datos y las tareas configuradas con su horario asociado por cada día de la semana.



Figura 13: Pantallas con la lista de tareas predefinidas y su relación con cada franja horaria

3.4 Diseño de la base de datos

Android usa *SQLite* como sistema de gestión de base de datos. *SQLite* es una biblioteca en la que sobre un archivo se almacena el esquema de base de datos y los propios datos. Esto hace que sea un sistema liviano de bases de datos y con un rendimiento óptimo, es decir, realiza operaciones más rápido que por ejemplo *MySQL*. En cuanto a la tipología es una base de datos relacional de fuente abierta, portable, multiplataforma que cumple los estándares *SQL92* y por lo tanto su sintaxis es *SQL*.

Uno de los atractivos de esta plataforma es que al ser empotrada no es necesario iniciar un servicio independientemente de la aplicación, es decir, es posible distribuir la aplicación sin que el cliente tenga la necesidad de configurar nada. Este conjunto de características hace de *SQLite* un sistema de gestión de base de datos idóneo para dispositivos móviles.

En la “Agenda comunicativa” es necesario almacenar información tanto de las tareas a realizar como del momento en el que deben ser realizadas. La figura 14 muestra el diagrama del modelo de la base de datos.

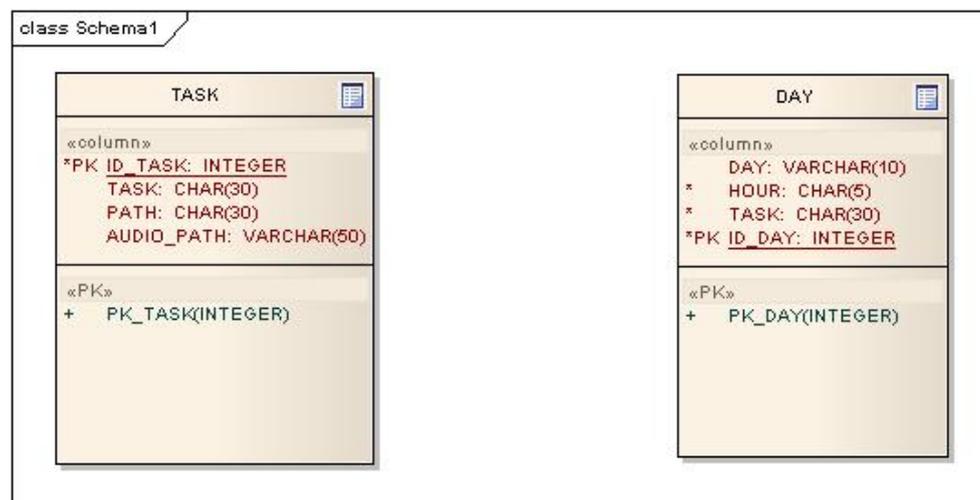


Figura 14: Modelo de base de datos

La base de datos consiste en dos tablas, una para representar las tareas en sí mismas y otra para almacenar las ocurrencias asociadas a cada día. Por un lado, la tabla de las tareas (“*task*”)



almacena información tanto de la ruta que contiene la imagen dentro del dispositivo como de la ruta que contiene el audio asociado con la imagen, así como el literal que especifica el contenido de la tarea. Por otro lado, la tabla sobre el momento en el que las actividades tienen que llevarse a cabo (“*day*”) guarda información relativa al día, hora y el literal de la tarea a realizar.

En concreto, la aplicación “*Agenda Comunicativa*” se instala en el dispositivo móvil con una serie de actividades precargadas en la tabla de “*task*” con el fin de facilitar las labores de configuración de nuevas tareas a los responsables de las personas autistas. Estas tareas, son las actividades de la vida cotidiana que deberán realizar este tipo de personas, como puede ser asearse, vestirse, peinarse, etc.

3.5 Implementación

Dentro de este apartado, se detalla el proceso de desarrollo de la aplicación, la implementación del mismo incluyendo capturas de pantalla del producto final.

3.5.1 Estructura

Tal y como se ha visto en el punto 2.4 de este documento, *Android* sigue el patrón “Modelo Vista Controlador” lo que implica lógicamente que “*Agenda comunicativa*” también lo implementa. Gracias a la instalación del *plugin* para *Android* en *Eclipse* (ver anexo A), es posible generar nuevos proyectos con un esqueleto predefinido para *Android*.

La estructura de cualquier aplicación definida para *Android* puede dividirse en los tres grandes grupos señalados en la figura 15:

- La carpeta “*src*” aloja el código fuente de la aplicación, desde la implementación de cada una de las pantallas “*com.diary.reference.screen*” hasta el paquete que incluye las clases de utilidades, pasando por las clases encargadas de la interacción con la base de datos “*com.diary.reference.manager*”.
- La carpeta “*res*” almacena los recursos de la aplicación como imágenes, iconos contenedores en *xml* de cada una de las pantallas encargados de definir la interfaz de usuario “*layout*” y “*values*” donde se definen las constantes como literales, definición de colores, definición de componentes de la interfaz de usuario así como su posicionamiento, etc.
- El fichero “*AndroidManifest.xml*” es vital para cada una de las aplicaciones de *Android* ya que contiene las configuraciones del proyecto, describe los componentes, cada una de las clases de cada componente, qué tipos de datos maneja, etc. Por ejemplo, en este fichero se define el paquete donde se encuentra la actividad de inicio de la aplicación, especificando también, como es obvio, la propia actividad que inicia la aplicación. Por otro lado, también se definen todas las actividades que forman parte de la aplicación, los permisos, etc. Al ser un archivo imprescindible, este archivo no puede cambiarse de nombre.

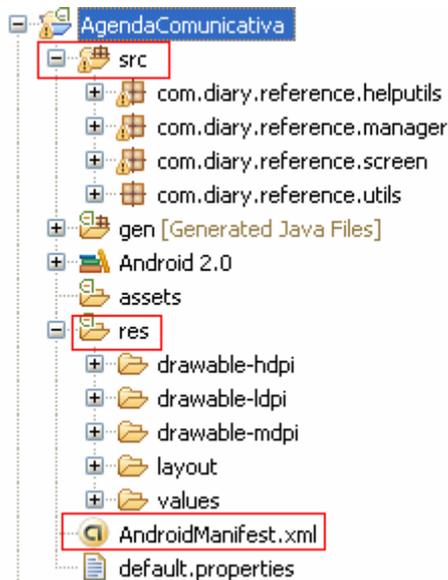


Figura 15: Esqueleto proyecto de “Agenda Comunicativa” en Eclipse

3.5.2 Implementación

Respecto a la implementación de las aplicaciones en *Android*, éstas se desarrollan siguiendo la abstracción “Actividades”. El flujo de la aplicación se va definiendo conforme se pasa de una actividad a otra pudiendo transferir información entre varias actividades.

En el caso concreto de la “*Agenda Comunicativa*”, se ha decidido desarrollar cada una de las pantallas de la aplicación como actividades independientes, es decir, cada una de las clases *Java* definen de cada una de las pantallas. Esta es la forma habitual, ya que encapsulas el comportamiento cada una de las pantallas en clases *Java* que heredan la funcionalidad de la clase “*Activity*”.

```
public class Agenda extends Activity {  
  
    public void onCreate(Bundle savedInstanceState) {  
  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
    }  
}
```

Figura 16: Actividad “Agenda”

Tal y como se observa en la figura 16, la clase “*Agenda*” hereda de la clase “*Activity*” definiendo el método *onCreate*. Este método se ejecuta en el momento en que se invoca esta actividad. Hay dos formas de invocarlo, desde otra actividad o definiéndola como actividad inicio de la aplicación. Por otro lado, *setContentView* define cual de las interfaces de usuario creadas (las cuales se encuentran en la carpeta *res.Layout*) es la que se corresponde con la actividad en cuestión.

Con esta configuración, es necesario definir un método tanto para pasar de una actividad a otra como para pasar datos de una pantalla a otra. Para facilitar la consecución de estos dos objetivos, *Android* provee la librería “*android.content.Intent*” donde se encuentran definidos métodos para cambiar de actividad y para intercambiar información entre actividades. A continuación se muestra un ejemplo donde la clase “*Config*” es la clase origen (clase que envía datos) y “*SelectTask*” es la clase destino de los datos a enviar (ver figura 17).

```
public class Config extends Activity {
    ...
    Intent intent = new Intent(Config.this, SelectTask.class);
    intent.putExtra("day", day);
    intent.putExtra("hour", hour);
    ...
}
public class SelectTask extends Activity {
    if (extras != null) {
        day = extras.getString("day");
        hour = extras.getString("hour");
    }
}
```

Figura 17: Intercambio de datos entre actividades.

En el código de la figura 17, se puede observar el envío de los datos “*day*” y “*hour*” desde la pantalla “*Config*” a la pantalla “*SelectTask*”. Para ello, en la clase “*Config*” se utiliza el método “*putExtra(key, value)*” donde el primero es el identificador del dato a enviar y el segundo es el contenido del mismo.

Por último, se muestra la estructura del fichero de configuración “*AndroidManifest.xml*”. Tal y como se ha comentado en el apartado anterior, este fichero es un componente vital para las

aplicaciones de *Android* ya que en él se definen tanto las actividades de la aplicación como la actividad de inicio de la aplicación. A continuación se muestra un extracto de este fichero de la aplicación “*Agenda Comunicativa*” (ver figura 18).

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.diary.reference.screen"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon"
        android:label="@string/app_name">
        <activity android:name=".Agenda"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".PlannerApp"
            android:label="@string/app_name">
        </activity>
        ...
        <activity android:name=".DailyTasks"
            android:label="@string/title_daily_tasks">
        </activity>
    </application>
    <uses-sdk android:minSdkVersion="5" />
</manifest>
```

Figura 18: Extracto del fichero *AndroidManifest.xml* de la ‘*Agenda Comunicativa*’.

Cabe destacar de este fichero de configuración, entre otras cosas, la definición del nombre de la aplicación (etiqueta “*application*” en el atributo “*android:label*”), la actividad principal (“*Agenda*”), la actividad de inicio de la misma (actividad “*Agenda*”), así como la definición de todas las actividades que forman parte de la aplicación “*Agenda Comunicativa*”.

3.5.3 Interfaz de usuario

Relacionado con la interfaz de usuario de las aplicaciones desarrolladas con *Android*, existen una serie de componentes predefinidos para la construcción del interfaz de usuario (en formato *xml*). En concreto, para la aplicación de la “*Agenda Comunicativa*” se ha hecho uso

de los componentes *textview*, *button*, *imagebutton*, *spinner* y *listview*. A continuación se detallan las características funcionales más importantes de cada uno de ellos.

TextView:

Es el contenedor encargado de mostrar literales de texto en las diferentes pantallas, definiendo distintas propiedades como el espacio que ocupa y el texto en sí mismo. Como se puede observar en el siguiente ejemplo (ver figura 19), el literal no aparece directamente ya que se definen dentro de contenedores de constantes. En este caso en concreto, el texto se define en “*res/values/strings*”. Esto conlleva ventajas como la internalización, ya que si se desea cambiar de idioma simplemente hay que modificar los literales de estos contenedores en lugar de modificar todos los puntos de la aplicación donde se hayan definido. Por ejemplo, si se desea mostrar la aplicación en francés, basta con añadir el contenedor “*res/values-fr/strings*”, de esta forma, si el móvil tiene por defecto el idioma francés la aplicación en cuestión aparecerá en este idioma.

```
<TextView  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"  
    android:text="@string/config1"/>
```

Figura 19: Componente UI *TextView*.

Button:

Componente visual que se corresponde con un botón al que se le proporciona la funcionalidad deseada. Dentro de este componente se especifica su identificador, el literal que aparece en el botón, y las dimensiones (ancho y alto) que ocupa en la pantalla. En el ejemplo de la figura 20, se representa el botón “Aceptar” incluido en varias pantallas dentro de la aplicación “Agenda Comunicativa”.

```
<Button  
    android:id="@+id/accept"  
    android:text="@string/accept"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
>
```

Figura 20: Componente UI *Button*.

En la figura 21, se muestra el componente “*TextView*” y el componente “*Button*” recuadrados en rojo. El primero se representa en la pantalla en forma de texto “SELECCIÓN DÍA A CONSULTAR” y el segundo en varios botones “Aceptar”, “Volver” y “Ayuda”.



Figura 21: Ejemplo de los componentes *TextView* y *Button*.

Image-Button:

Componente de *Android* con la misma funcionalidad que el anterior. La diferencia es que es una imagen la que adquiere el rol de botón. La figura 22 muestra el código de este botón con imagen y la figura 23 un ejemplo de uso de la misma.

```
<ImageButton  
    android:background="@null"  
    android:id="@+id/image"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"  
    android:src="@drawable/clockred" />
```

Figura 22: Componente UI *ImageButton*.



Figura 23: Ejemplo componente *Imagebutton*

Spinner:

Android provee diferentes componentes para facilitar la elección entre diferentes datos seleccionables. Entre ellos, se barajó la posibilidad de usar los componentes *radiobutton*, *checkbox* pero el que más se ajusta a las necesidades de “*Agenda Comunicativa*” es el componente “*Spinner*”. Tal y como se observa es la figura 24, es posible definir el identificador del componente dentro de la aplicación, el ancho y alto que ocupará en la pantalla, si se mostrará un título al desplegable y cuál será el contenido del mismo. Esto último se define dentro del contenedor de *arrays* por defecto de *Android* “*res/values/arrays*” (ver figura 25), lo cual conlleva ventajas como la internalización.

```
<Spinner
    android:id="@+id/spinner"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:drawSelectorOnTop="true"
    android:prompt="@string/day_prompt" />
```

Figura 24: Componente UI Spinner.

```
<resources>
...
<string-array name="days">
    <item>Lunes</item>
    ...
    <item>Domingo</item>
</string-array>
...
</resources>
```

Figura 25: Contenido de "res/values/arrays".

Este componente es empleado en varias pantallas de la aplicación, en concreto para la selección de uno de los días de la semana y la selección de la hora del día.

En la figura 26, se muestran los dos *spinners* comentados en donde el usuario final puede seleccionar el día y la hora. Como se puede observar, ambos componentes se definen en su parte superior con el título del desplegable.



Figura 26: Spinners para la selección de los días de la semana y la hora de realización de una tarea.

Dentro de este componente se le ha añadido un *listener* de tal forma que una vez seleccionado el ítem se mostrará un pequeño mensaje en pantalla del mismo. Se trata de una clase que implementa “*OnItemSelectedListener*”, la cual es un *listener* que captura los eventos de selección. En este caso, la selección de uno de los elementos del desplegable. La acción a tomar cuando se selecciona un elemento en la “*Agenda comunicativa*” (ver figura 27) es mostrar un mensaje en forma de ventana emergente sobre el elemento seleccionado.

```
public class DaySelectedListener implements OnItemSelectedListener {  
    public void onItemSelected(AdapterView<?> parent,  
        View view, int pos, long id) {  
        Toast.makeText(parent.getContext(), "El día seleccionado es: " +  
parent.getItemAtPosition(pos).toString(), Toast.LENGTH_LONG).show();  
        day = parent.getItemAtPosition(pos).toString();  
    }  
  
    public void onNothingSelected(AdapterView parent) {}  
}
```

Figura 27: Código de un *listener* que muestra un mensaje en forma de ventana emergente

La figura 28 muestra un ejemplo del mensaje en forma de ventana emergente de la selección de un día dentro del desplegable de días de la semana, siendo el día seleccionado por el usuario el viernes.



Figura 28: Mensaje en forma de ventana emergente.

ListView:

El objetivo principal de este componente es listar una serie de elementos en pantalla, definiendo el identificador del listado y el tamaño (ancho y alto) que ocupará en la pantalla. Este componente es alimentado en forma de *array* de literales que serán los que se listen en la pantalla de la aplicación. Un ejemplo del código se puede ver en la figura 29.

```
<ListView android:id="@android:id/list"  
          android:layout_width="wrap_content"  
          android:layout_height="wrap_content"/>
```

Figura 29: Componente de la interfaz de usuario *ListView*.

Para la “Agenda Comunicativa” se hace uso de este elemento para listar las diferentes tareas disponibles (ver figura 30) y listar las tareas registradas para cada día de la semana.

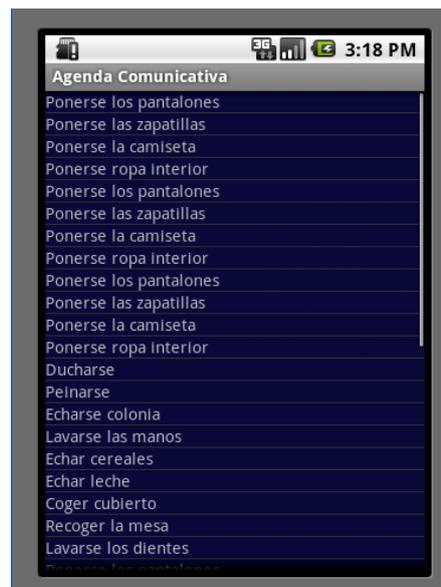


Figura 30: *ListView* de las tareas disponibles

Por otro lado, aunque no sea un componente de la interfaz de usuario como tal, también se ha usado la biblioteca “*android.app.AlertDialog*” gracias a la cual es posible mostrar en las pantallas de la aplicación pantallas en forma de ventanas emergentes. Para ello hay que crear una instancia de la clase *AlertDialog* invocando varios de sus métodos para establecer los valores de sus distintos atributos como el título, el mensaje a mostrar, la imagen asociada y el botón de cierre de la ventana emergente.

En concreto para la aplicación “*Agenda Comunicativa*” se ha usado para mostrar dos tipos de mensajes: uno muestra la ayuda en forma de texto de cada una de las pantallas (figura 31) y el otro muestra una ventana emergente solicitando la confirmación o no de salida de la aplicación (figura 32).

```
/** Método encargado de mostrar la ayuda de esta pantalla */  
private void showHelp(int title, int message, int icon, int butto){  
    new AlertDialog.Builder(this)  
        .setTitle(title)  
        .setMessage(message)  
        .setIcon(icon)  
        .setNeutralButton(button_text,  
            new DialogInterface.OnClickListener(){  
                public void onClick(DialogInterface dialog,  
                    int whichButton) {  
                }  
            }).show();  
}
```

Figura 31: Código que muestra la ayuda en forma de ventana emergente.

En la figura 32, se indica el código encargado de mostrar al usuario la pantalla de verificación de salida de la aplicación. Como en el caso anterior se crea una instancia de *AlertDialog*, definiendo en este caso dos botones: uno para confirmar la salida de la aplicación y otro para cancelar esta acción, volviendo a la pantalla desde la cual se ha lanzado la acción de salida.

```
/** Método encargado de mostrar el mensaje de salida  
 * @param message  
 * @param positive_button  
 * @param negative_button  
 */  
private void AskOut(int message, int positive_button, int  
negative_button) {  
    new AlertDialog.Builder(this)  
        .setMessage(message)  
        .setCancelable(false)  
        .setPositiveButton(positive_button, new  
DialogInterface.OnClickListener() {  
            public void onClick(DialogInterface dialog, int id) {  
                Config.this.finish();  
            }  
        })  
        .setNegativeButton(negative_button, new  
DialogInterface.OnClickListener() {  
            public void onClick(DialogInterface dialog, int id) {  
                dialog.cancel();  
            }  
        })  
        .show();  
}
```

Figura 32: Código para pedir confirmación de salida de la aplicación en forma de ventana emergente.

Por último, la figura 33 muestra los tipos de ventanas emergentes, explicados anteriormente en las figuras 31 y 32 aplicadas en la “Agenda Comunicativa”. En la parte izquierda de la figura, se muestra la ayuda en forma de texto de la pantalla desde la cual se ha solicitado. En la parte derecha, se pide la confirmación o cancelación de la salida de la aplicación.



Figura 33: *AlertDialog* correspondientes al botón *Ayuda* y al botón *Salir* respectivamente.



3.6 Escenarios

Una vez detallado el análisis, el diseño y la implementación de esta aplicación, es el momento de describir los posibles escenarios de la aplicación *Agenda Comunicativa*. Esta descripción de escenarios, está dirigido al padre, responsable o tutor encargado del cuidado de la persona con autismo o TGD con el fin, tanto de sacar el máximo partido de la aplicación como de resolver cualquier duda o problema que surja del uso de la misma. Para ello, se describirá de forma detallada y concisa las diferentes funcionalidades que ofrece la aplicación.

Para la correcta comprensión de los distintos escenarios se ha decidido separar la descripción en dos partes: escenario para el rol de persona con autismo o TGD y escenario para el rol de tutor, padre o responsable.

3.6.1 Escenario con el rol padre, responsable o tutor

Para una composición del lugar idónea, primero se describe el escenario con el rol padre, tutor o responsable ya que es el encargado de configurar las diferentes tareas de la agenda que será consultada posteriormente por el niño con autismo. Se describe en forma de capturas de pantalla de la propia aplicación paso a paso los diferentes flujos posibles.

Etapa 1: Selección de la aplicación *Agenda Comunicativa*.

Para acceder a la aplicación, es necesario acceder al menú principal de aplicaciones de *Android* (parte izquierda de la figura 34). Desde esta pantalla, se selecciona el botón “*Aplicaciones para personas con necesidades especiales*” y se accede a la pantalla de la parte derecha de la figura 34. En esta pantalla se muestra el menú de inicio de las aplicaciones disponibles para personas con necesidades especiales. En concreto, aparecen dos aplicaciones, la que se refiere a este documento “*Agenda Comunicativa para personas con TGD o autismo mediante dispositivos móviles*” y la aplicación “*Dispositivo de ayuda a la comunicación, para personas con discapacidad auditiva, sin intérprete, u oyentes con desconocimiento del idioma del receptor*” realizada por Jebel Gil, la cual facilita la comunicación entre el emisor y receptor en escenarios especiales como el caso en que alguno de los interlocutores tiene

problemas de audición o desconoce el idioma. Una vez situados en la pantalla de la parte derecha de la figura 34, pulsaremos sobre el botón “Agenda Comunicativa” para acceder a la aplicación.

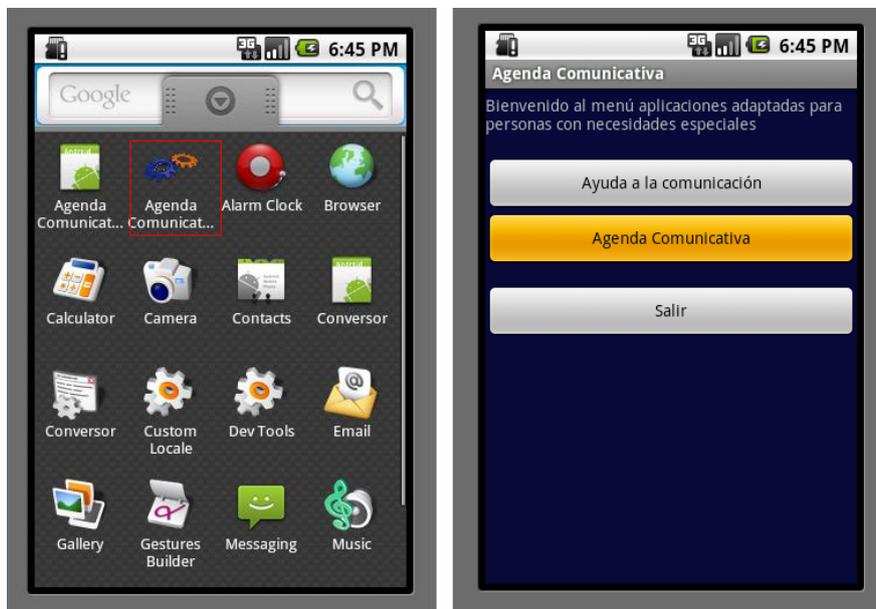


Figura 34: Menú de aplicaciones de Android y menú de aplicaciones para personas con necesidades especiales

Etapa 2: Pantalla inicio *Agenda Comunicativa*.

Una vez completada la primera etapa 1, la aplicación muestra la pantalla situada en la parte izquierda de la figura 35, donde es posible consultar la tarea a realizar justo en el momento de la consulta pulsando sobre la imagen del reloj🕒. Además de esto, es posible salir de la aplicación pulsando sobre la imagen-botón🔴 situada en la parte inferior izquierda de esta pantalla. Si se pulsa sobre el botón-imagen de la parte inferior derecha🌐, se accede a la pantalla de configuración de la agenda (ver parte derecha de la figura 35).



Figura 35: Pantalla de inicio de la *Agenda Comunicativa* y configuración de tareas

Etapa 3: Registro de tareas.

El registro de tareas (ver figura 35 parte derecha) en la agenda la realiza el padre, responsable o tutor a través del botón inferior derecho  que se muestra en la parte izquierda de la figura 35. La pantalla de gestión de tareas se divide en dos:

- La parte superior permite seleccionar el día y la hora de la tarea a registrar
- En la parte inferior reside la funcionalidad de consultar y eliminar tareas comentada en detalle en el siguiente apartado (etapa 4: Manejo de tareas) de este documento.

Aparte de seleccionar el día y la hora de la tarea (ver figura 36), en la parte inferior de la pantalla (parte derecha de la figura 35) se muestran dos botones mediante los cuales es posible volver a la pantalla anterior o salir de la aplicación.

Una vez seleccionado el día y la hora de la tarea a registrar se pulsa sobre el botón “Aceptar”, cuya acción provoca que se muestre toda la lista de posibles tareas. Para registrar una de ellas basta con pulsar sobre la tarea que se desea almacenar (ver figura 37).



Figura 36: Pantalla de selección de día y hora.



Figura 37: Pantalla de selección de una de las tareas disponibles

Etapa 4: Manejo de tareas

Desde la pantalla *Configuración de agenda* (parte derecha de la figura 35) se puede gestionar las distintas tareas consultando las tareas definidas hasta el momento o eliminar tareas de un determinado día.

Para consultar las tareas de un día, se pulsa sobre el botón *Consulta tareas/día*. La aplicación muestra la pantalla de la parte izquierda de la figura 38. El usuario seleccionará el día en el desplegable pulsando a continuación el botón *Aceptar*. A continuación se mostrará una pantalla similar a la de la parte derecha de la figura 38. En esta pantalla se pueden ver las tareas registradas para el día escogido además de los botones *Volver* y *Ayuda*, los cuales retornan a la pantalla anterior y la pantalla de ayuda respectivamente.

En cuanto a la eliminación de tareas, existen dos posibilidades: eliminar una tarea individual y eliminar todas las tareas de un determinado día. La primera es posible mediante la pantalla de la parte derecha de la figura 37, en la que es necesario seleccionar la tarea registrada del menú superior que se desea eliminar y pulsar a continuación el botón *Eliminar tarea*.

En cuanto a eliminar tareas por días, es necesario situarse en la pantalla de la parte derecha de la figura 35 y pulsar a continuación el botón *Eliminar tareas/día*. Inmediatamente se mostrará la pantalla de la figura 38 donde el usuario deberá seleccionar un día del menú desplegable. Una vez seleccionado, se pulsará el botón *Aceptar*. Antes de efectuar el borrado se preguntará al usuario a través de una ventana emergente la confirmación de esta acción.



Figura 38: Selección del día a consultar y tareas registradas de un día



Figura 39: Pantalla selección eliminar tarea de un día determinado

Pantallas de salida y de ayuda

Por último, se muestran las pantallas que se corresponden a pulsar sobre los botones “Ayuda” (parte izquierda de la figura 40) y “Salir” (parte derecha de la figura 40) que se repiten a lo largo de la aplicación.



Figura 40: Pantalla de ayuda y pantalla de confirmación de salida de la aplicación

3.6.2 Escenario para una persona con autismo o TGD

Una vez configurada las tareas por parte del responsable, tutor o padre, se va a describir un posible escenario en el que una persona con autismo o TGD interactúa con la aplicación desarrollada de la *Agenda Comunicativa*. Por ejemplo, imaginemos que es un lunes por la mañana a las 8:00 am. En este momento la persona autista se despierta, desayuna su plato de cereales habitual y su vaso de leche. Después se peina. A continuación, consulta la aplicación “Agenda Comunicativa” para ver qué tarea tiene que realizar en este momento (ver figura 41). La persona autista sólo tiene que pulsar sobre la imagen-botón del reloj (parte izquierda de la figura 41) para ver la tarea actual. Acto seguido, se muestra la pantalla de la parte derecha de la figura 41 en la cual se visualiza en la parte principal de la pantalla, la imagen real de la tarea a realizar previamente configurada para este momento de la semana. Pulsando sobre el botón-imagen  se puede visualizar el resto de tareas programadas para ese tramo horario. Además, en la parte superior de la pantalla se visualizan en un tamaño reducido las tareas que se han tenido que realizar anteriores a la actual (en este escenario, el desayuno de los cereales y el vaso de leche junto con la tarea de peinarse). Con esto, se consigue no perder el hilo de tareas en el caso de no poder haberse realizado con anterioridad. Por último indicar que existe una ayuda en formato de audio, mediante la cual se reproduce en forma de audio una explicación de la tarea en curso. Esta ayuda se reproduce pulsando sobre el botón .

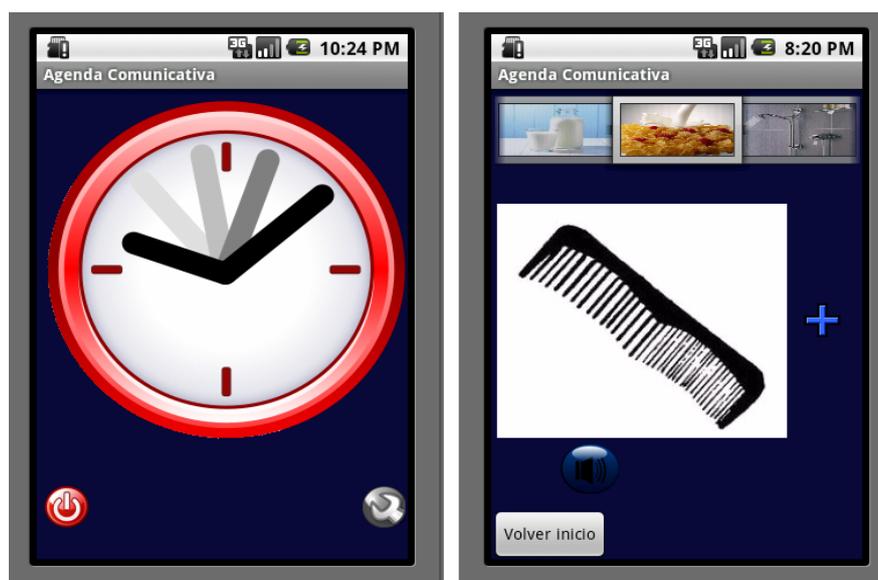


Figura 41: Pantalla de inicio y consulta de la tarea actual para una persona autista



**Agenda comunicativa para personas con TGD o
autismo mediante dispositivos móviles**

Tutora: Estefanía Martín
Alumno: Miguel A. Santiago



4. CONCLUSIONES Y POSIBLES AMPLIACIONES

En el apartado de motivación de este documento se indicaba que el objetivo de la aplicación “*Agenda Comunicativa*” era desarrollar una aplicación que intente satisfacer las necesidades de personas con TGD o autismo en el campo de la comunicación y planificación del día a día de este tipo de usuarios. Siendo esto un complemento de las terapias actuales en las que uno de los puntos importantes es la organización de la vida diaria de estos pacientes.

Éste sin duda es el logro principal, en el cual la aplicación desarrollada de la “*Agenda Comunicativa*” permite mejorar la calidad de vida tanto de las personas con autismo como de sus responsables, pero evidentemente no es el único. Está unido a otros logros importantes por parte del alumno como el aprendizaje de un *framework* como *Android* y las diferentes etapas a seguir para el desarrollo de una aplicación informática para dispositivos móviles. Estas etapas comprenden desde la presentación de la propuesta y desarrollo de una evaluación correcta de alternativas hasta la implementación de la misma, pasando por el proceso de análisis y desarrollo. En todas ellas, se realiza un análisis de herramientas específicas de cara a facilitar el trabajo y obtener el resultado óptimo.

Para este desarrollo se ha hecho uso de una metodología de desarrollo iterativa que abarca la planificación hasta la documentación haciendo uso de diagramas UML. De esta forma se ha construido la aplicación de forma incremental cumpliendo hitos y adaptando la interfaz gráfica para afianzar la usabilidad de cara al usuario final (personas con TGD) hasta obtener el producto final.

Por otro lado, otro aspecto importante ha sido el aprendizaje de patrones de diseño *software*. En concreto, el “Modelo Vista Controlador”, el cual es usado para el desarrollo de aplicaciones de terceros de *Android*. De este modo la “*Agenda comunicativa*” se beneficia de las ventajas de este patrón obteniendo un código estructurado que facilita la labor de mejoras y/o mantenimientos.

En cuanto al uso de herramientas para el desarrollo de aplicaciones informáticas, indicar el aprendizaje de entornos de desarrollo integrado como Eclipse, entornos de programación Java y el refuerzo de conocimientos en el metalenguaje XML.



Todo este proceso detallado anteriormente, propuesta de la aplicación, captura de requisitos, estudio de alternativas de plataformas de desarrollo, diseño, implementación, documentación y plan de pruebas ha conllevado un coste total de 450 horas.

Respecto a la continuidad del proceso implementación de nuevas funcionalidades de la aplicación “Agenda Comunicativa”, se podrían realizar diferentes mejoras. Todas estas mejoras estarían enfocadas a la interacción y experiencia adquirida de los diferentes usuarios finales. Algunas de estas mejoras serían las siguientes:

- Adaptación de la interfaz de usuario de la aplicación dependiendo del grado de autismo del usuario final, es decir, la interfaz tendrá unas características u otras dependiendo de su grado de autismo, facilitando todavía más la interacción a las personas con un grado de autismo alto.
- Registro de nuevas tareas diferentes a las precargadas por parte de usuarios con el rol padre, responsable o tutor. En concreto en el caso de dispositivos móviles que dispongan de cámara de fotos, realizar una foto a la actividad a registrar e importarla a la base de datos de la aplicación “*Agenda Comunicativa*”.
- Conexión a una base de datos remota, contra la cual se pudiera descargar un programa de tareas específico para cada caso de autismo el cual haya sido previamente configurado por un terapeuta.
- Inclusión de pantallas para aumentar la comunicación de la persona autista desde las cuales la persona que padece autismo o TGD, fuera capaz de transmitir sus sentimientos respecto a la realización de la tarea en curso.

Como se puede observar, las mejoras dependen en gran medida de la experiencia de usuario, tanto del usuario final con rol persona con TGD como del usuario con el rol de padre, responsable, tutor o terapeuta. También, la figura del terapeuta toma especial valor, ya que es capaz de aportar la funcionalidad adecuada de cara a construir una agenda completa para personas con este tipo de discapacidades.



5. BIBLIOGRAFÍA

Libros:

- Pérez Neira, L., Guillén Rubio, A., Pérez Martínez, M.I., Jiménez León, I., Bonilla Barragán M.J.: “La atención educativa al alumnado con trastorno del espectro autista.” Junta de Andalucía. Disponible en: http://www.cepalcala.org/upload/recursos/16_09_08_11_30_24.pdf.
- Nicolas Gramlich, “Handbook! Android Programming”. Disponible en <http://andbook.anddev.org>
- Anthony Stevens, Mark Murphy, Gabor Paller “Hello, Android”. Ediciones Burnette.

Sitios web:

- El Servicio de Información sobre Discapacidad (SID): <http://sid.usal.es>
- “MedlinePlus”, servicio de la Biblioteca Nacional de Medicina de EE.UU y los institutos nacionales de la salud: <http://www.nlm.nih.gov>
- Sitio web de la Dra. Maria Jesus Clavera Ortiz y del Dr. Javier Martinez Ruiz, sobre Trastornos del Desarrollo: <http://www.medicinadeldesarrollo.com>
- Psicología de la educación para padres y profesionales: <http://www.psicodiagnosis.es>
- Psicología Clínica Infanto-Juvenil: www.psicopedagogia.com
- Grupo de desarrollo en sistemas *Android*: <http://www.desarrolladores-android.com>
- Foro especializado en desarrollo y explotación de aplicaciones *Android*: <http://www.android-spa.com>
- Sitio web oficial de Google con todo lo necesario para el desarrollo de aplicaciones *Android*: <http://developer.android.com/index.html>
- Sitio web oficial de la base de datos SQLite: <http://www.sqlite.org>

Aplicaciones:

- Balsamiq Mockups: Aplicación para el desarrollo de bocetos de la implementación <http://www.balsamiq.com/>
- Eclipse Galileo versión 3.5.1: entorno de desarrollo integrado (IDE) de código abierto.



- Android Development Tools (ADT): *plugin* del IDE Eclipse, el cual, ofrece un entorno integrado para el desarrollo de aplicaciones para *Android*.
- Audacity: Programa libre y código abierto utilizado para grabar y editar sonido.
- Paquete Microsoft Office.



6. ANEXO: Configuración del entorno de desarrollo

En el presente anexo se detalla tanto los componentes *software* como la configuración del entorno de desarrollo necesario para implementar aplicaciones en *Android*.

I - Software

Los componentes *software* que se han utilizado para el desarrollo de la aplicación *Agenda comunicativa* han sido:

- IDE: Eclipse versión Galileo 3.5.1
- JDK versión 1.6
- Android SDK versión 2.0
- Base de datos: SQLite

II - Instalación del entorno de desarrollo

Queda fuera del ámbito de este anexo la instalación del IDE Eclipse y los componentes de desarrollo JDK, ambos necesarios para la implementación de este proyecto. Este apartado se centra en el *software* necesario para poder implementar aplicaciones en *Android*.

Instalación de la SDK:

1. Descargar el SDK desde la web oficial de *Android*:
<http://developer.android.com/index.html>.
2. Descomprimir el fichero en la ruta deseada.
3. Añadir la ruta anterior a la variable de sistema "PATH" definida dentro de las variables de entorno (p.e. En Windows, mi PC → Propiedades de sistema → Variables de entorno). La figura 41 muestra un ejemplo de variables de entorno de Windows. La figura 42 muestra cómo actualizar la variable de entorno "PATH" para utilizar el SDK de *Android*.



Figura42: Variables del sistema en Windows



Figura 43: Edición de la variable PATH en Windows

Instalación del plugin para Eclipse:

Una vez iniciado Eclipse, hay que situarse en el menú *Help*. Seleccionar la opción “*Install New Software*” y pulsar sobre el botón “*Add*”. En este momento se mostrará una pantalla análoga a la de la figura 43 donde se pide el nombre y la URL del lugar donde se desea descargar el *plugin*.

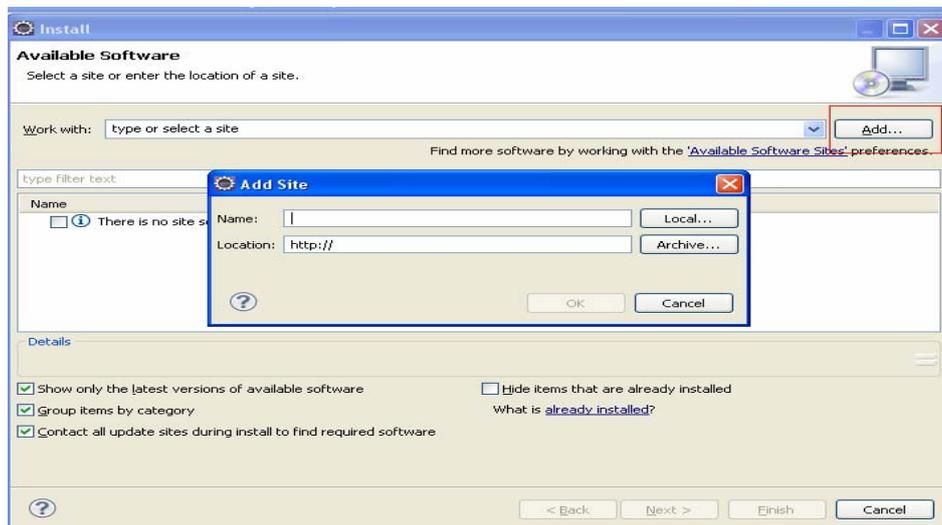


Figura 44: Pantalla de Eclipse para añadir el plugin de Android

Añadimos la siguiente dirección: <https://dl-ssl.google.com/android/eclipse/>. Una vez completados los pasos anteriores se está en disposición de implementar aplicaciones para *Android*.