



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
Escuela de Ingeniería Informática



ESCUELA EN INGENIERÍA INFORMÁTICA

TRABAJO DE FIN DE GRADO

GRADO DE INGENIERÍA INFORMÁTICA

**Elaboración de prototipos para el desarrollo de
productos de tecnología social: ProMat5**

Tutora: Dra. Beatriz Correas Suárez
Profesora Titular de Universidad
Departamento de Informática y Sistemas

Autor: Moisés Hernández Falcón

Las Palmas de Gran Canaria – Julio 2014

Dedicatoria

Este Trabajo Fin de Título se lo dedico a toda mi familia por haber estado apoyándome durante toda este proceso formativo en la Universidad de Las Palmas de Gran Canaria, ya que sin ellos y su apoyo; tanto económico como emocional; nada de esto hubiese sido posible.

No quisiera dejar de recordar y hacer mención a todas aquellas personas (tanto a las que han llegado al final del camino conmigo, como a aquellas que no lo han conseguido) que han estado a mi lado en mi día a día en la Facultad, con quienes comencé mi camino en el ámbito universitario y sin los cuales hubiese sido más difícil mi andadura universitaria.

Índice

Introducción	Páginas 4 - 5
Estado Actual y Objetivos	Páginas 6 - 7
Justificación de las competencias	Páginas 8 - 9
Normativa y Legislación Europea	Páginas 10 - 11
Normativa y Legislación Nacional	Páginas 12 - 16
Menores y Protección de Datos	Páginas 17 - 20
Aportaciones	Página 21
Análisis	Páginas 22 - 43
Requisitos Software y Hardware	Páginas 44 - 45
Desarrollo	Páginas 46 - 50
Diseño	Páginas 51 - 60
Conclusiones	Página 61
Bibliografía	Página 62

Introducción

La **Tecnología Social** es un movimiento social espontáneo que responde a una nueva línea de pensamiento popular, que une a todas las razas y culturas del mundo tras ideales y proyectos sociales, que van más allá de las fronteras políticas o religiosas. El principal eje es la comunicación, que surge por la necesidad vital de crear una situación, o un estado de las cosas, distinto del que se vive en la actualidad.

Como fin, pretende fomentar el desarrollo de hacer algo importante de la tecnología de sus bienes y aplicación de conocimientos y tecnologías con fines netamente sociales, absolutamente pacíficos y opuestos a los objetivos comerciales o militares que determinan el actual avance y desarrollo tecnológico de la humanidad.

Este movimiento busca establecer una nueva escala de valores éticos y sociales, apuntando al estado de bienestar universal, administrado por instituciones populares apolíticas denominadas foros, donde todos los miembros de la sociedad, son administradores directos del patrimonio social, sin intermediarios ni representaciones políticas de ninguna índole.

A fines del siglo XX, con el auge de las comunicaciones y la aparición de Internet como medio de expresión realmente libre, por primera vez en la historia de la humanidad, la gente común pudo acceder por completo al conocimiento, a las ciencias y a las técnicas que pueden generar un estado de bienestar universal. Este libre acceso al conocimiento, actualmente está generando una nueva línea de razonamiento popular, que el actual uso comercial indiscriminado, y las aplicaciones bélicas de las complejas tecnologías disponibles, pueden y están generando daños irreparables a la humanidad.

La tecnología social podría interpretarse así: “Conjunto de teorías y de técnicas que permiten el aprovechamiento práctico del conocimiento científico con lenguaje propio de una ciencia o arte perteneciente o relativa a la sociedad.”

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

Como podemos apreciar, el sentido literal de la frase resultante es un tanto abstracto, no obstante, en la expresión “tecnología social” encontramos varios puntos muy claros para todos los que la interpretamos...

- En todos los casos se trata de “poner a disposición de la sociedad”, los conocimientos y avances científicos.
- Todos asumimos que la ciencia y el conocimiento, pueden facilitar el acceso al bienestar social.
- Por último, todos coincidimos en que la expresión “Tecnología Social” tiende a fomentar el desarrollo tecnológico con fines netamente sociales.

Estado Actual y Objetivos

Las tecnologías aplicadas a la educación suponen un beneficio en el proceso de enseñanza y aprendizaje, sobre todo cuando está orientado a edades tempranas y se combina el aspecto lúdico con el académico.

Esto sucede con las aplicaciones desarrolladas para tabletas, en donde los niños se encuentran cómodos, por su similitud con los juegos electrónicos a los que están acostumbrados, gracias a que las nuevas tecnologías móviles están cada vez más integradas en nuestra sociedad.

Pero no todas las aplicaciones desarrolladas se adaptan a las competencias que un estudiante debe adquirir en su educación primaria. En nuestro caso, gracias a la Tecnología Social, se está elaborando en colaboración con la Consejería de Educación, un estudio de dichas tecnologías y analizando las necesidades tecnológicas que van apareciendo.

Concretamente, para el caso de las Matemáticas, encontramos numerosas aplicaciones orientadas al aprendizaje de operaciones básicas: suma, resta, multiplicación, división, ... pero ninguna que enseñe a los niños a aplicar dichas operaciones para la resolución de problemas.

Y este es el fin de este Trabajo Fin de Grado; diseñar un prototipo de aplicación orientada a Tablet, con tecnología Android, para la resolución de problemas matemáticos para alumnos de niños de 5º de primaria, con tres características fundamentales:

1. La posibilidad de disponer de un espacio, en el cual los alumnos puedan realizar los cálculos necesarios para solucionar los problemas.
2. La posibilidad de incorporar un mecanismo, gracias al cual los usuarios puedan aportar nuevos problemas matemáticos, a modo de feedback con el usuario.
3. La combinación del aspecto lúdico con el académico

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

El principal objetivo de este Trabajo Fin de Grado es el de : desarrollar un prototipo orientado hacia dispositivos Tablet basados en Android, que permitan el aprendizaje sobre la materia de Matemáticas para un nivel de alumnos de 5° de primaria; y que cumpla con las características fundamentales anteriormente descritas.

Justificación de las competencias específicas cubiertas

CII01

La capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación vigente.

Esta competencia queda cubierta en los apartados estado actual y objetivos del Trabajo de Fin de Grado, y en los requisitos hardware y software. En ellos se especifican, por un lado, la metodología de diseño empleada, los motivos de diseño, desarrollo, etc, ... explicando exhaustivamente las fases de análisis y diseño de los casos de uso necesarios para satisfacer las necesidades de la aplicación, y por otro lado se justifica la utilización del hardware más adecuado para la implementación de la misma, teniendo en cuenta parámetros de fiabilidad, calidad y seguridad.

En resumen, con esta elección de hardware y software, la implementación de este trabajo de fin de grado da como resultado una aplicación robusta y conforme a los principios éticos y a la legislación vigente.

CII02

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

De todos los elementos de un sistema de gestión de proyectos, la planificación es posiblemente la más importante, puesto que una mala o inexistente planificación conducirá a una mala realización en el proyecto, lo cual repercutirá enormemente tanto en la calidad como en el plazo de terminación del proyecto, ocasionando grandes perjuicios. Esta es la razón por la que esta competencia queda cubierta, puesto que la buena planificación de este trabajo de fin de grado ha dado lugar al propósito fundamental del mismo, que no es más que la finalización en los plazos establecidos así como conseguir de manera adecuada los objetivos propuestos inicialmente.

CII18

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

Un análisis de las legislaciones que se han promulgado en diversos países arroja que las normas jurídicas que se han puesto en vigor están dirigidas a proteger la utilización abusiva de la información reunida y procesada mediante el uso de computadoras, e incluso en algunas de ellas se ha previsto formar órganos especializados que protejan los derechos de los ciudadanos amenazados por los ordenadores.

Desde hace aproximadamente diez años la mayoría de los países europeos han hecho todo lo posible para incluir dentro de la ley, la conducta punible penalmente, como el acceso ilegal a sistemas de computo o el mantenimiento ilegal de tales accesos, la difusión de virus o la interceptación de mensajes informáticos.

En la mayoría de las naciones occidentales existen normas similares a los países europeos. Todos estos enfoques están inspirados por la misma de preocupación de contar con comunicaciones electrónicas, transacciones e intercambios tan confiables y seguros como sea posible.

Dar un concepto sobre delitos informáticos no una labor fácil y esto en razón de que su misma denominación alude a una situación muy especial, ya que para hablar de "delitos" en el sentido de acciones tipificadas o contempladas en textos jurídico-penales, se requiere que la expresión "delitos informáticos" este consignada en los códigos penales, lo cual en nuestro país, al igual que en muchos otros, no ha sido objeto de tipificación aún; sin embargo, muchos especialistas en derecho informático emplean esta alusión a los efectos de una mejor conceptualización.

Normativa y regulación de la informática en el ámbito europeo

Hasta ahora, el principal esfuerzo europeo por regular el tema de los delitos informáticos dio como resultado el “Convenio sobre la Ciberdelincuencia”, de 21 de noviembre de 2001. Este documento fue firmado por los representantes de cada país miembro del Consejo de Europa, aunque su eficacia depende de su posterior refrendo por los órganos nacionales de cada país firmante.

El “Convenio sobre la Ciberdelincuencia” permitió la definición de los delitos informáticos y algunos elementos relacionados con éstos, tales como “sistemas informáticos”, “datos informáticos”, o “proveedor de servicios”.

Estos delitos informáticos fueron clasificados en cuatro grupos:

1º Delitos contra la confidencialidad, la integridad y la disponibilidad de los datos y sistemas informáticos.

- Acceso ilícito a sistemas informáticos.
- Interceptación ilícita de datos informáticos.
- Interferencia en el sistema mediante la introducción, transmisión, provocación de daños, borrado, alteración o supresión e estos.
- Abuso de dispositivos que faciliten la comisión de delitos.

2º Delitos informáticos.

- Falsificación informática que produzca la alteración, borrado o de datos informático que ocasionen datos no auténticos.
- Fraudes informáticos.

3º Delitos relacionados con el contenido.

- Delitos relacionados con la pornografía infantil.

4º Delitos relacionados con infracciones de la propiedad intelectual y derechos afines.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

Conviene destacar que en el “Convenio sobre la Ciberdelincuencia” se encomienda a cada Parte que tome las medidas necesarias para tipificar como delito en su derecho interno cada uno de los apartados descritos en cada categoría.

En la Disposición 14221 del BOE núm. 226 de 2010, encontramos el Instrumento de Ratificación del Convenio sobre la Ciberdelincuencia, hecho en Budapest el 23 de noviembre de 2001.

Normativa y regulación de la informática en el ámbito nacional

Leyes y Decretos Ley.

Ley Orgánica de Protección de datos de carácter personal: régimen sancionador aplicable (BOE no298 de 14/XII/99 que publicó la Ley Org. 15/1999 de 13 de Dic.)

Objeto: Proteger y garantizar las libertades públicas y derechos fundamentales de las personas, especialmente su HONOR e INTIMIDAD personal y familiar.

Aspectos de interés: Serán responsables: "Los responsables de los ficheros o de los tratamientos" y "los encargados de los tratamientos"

Tipos de Infracciones:

- Leves (art.44.2): multas de 100.000 a 10M pts. Ej. Rectificar datos o no comunicarlos a la Agencia de Protección Datos.

- Graves (art.43): multas de 10M a 50Mpts. Ej. No mantener sistemas de seguridad, obstrucción a inspección, uso en provecho propio...

- Muy Graves (art.45): multas de 50M a 100M ("Conductas reprochables") Ej. Vulnerar a propósito el secretismo, etc...

Ley 7/1998 de 13 de Abril que regula las condiciones generales de contratación.

R.D. 1906/1999 de 17/XII que regula la contratación telefónica.

R.D. Ley 14/1999 de 17/XII sobre Firma Electrónica (BOE No224 de 18/XII)

Firma electrónica: Dispositivo electrónico que permite la identificación del signatario de las operaciones realizadas por Internet.

Identifica: El firmante (autenticación) y Evita el retracto (no repudio).

Código Penal

Ley Orgánica 10/1995 de 23/XI

Tipifica delitos y faltas por el uso de la informática, concretamente contra la Intimidad, Patrimonio, Socioeconómicos y Propiedad Intelectual.

Título X: "Delitos contra la intimidad, derecho a la propia imagen y la inviolabilidad del Domicilio".

-Apoderarse de papeles, e-mails, mensajes, otros...

-Cracks: delitos

-Obtener datos de terceros...

Recomendaciones de la Agencia de Protección de Datos

Información en la recogida de datos

Cuando suministre datos personales a cualquier organización (proveedores de contenido, vendedores a través de comercio electrónico, etc.) sea consciente de a quién se los facilita y con qué finalidad.

Procure averiguar la política de sus proveedores y administradores de listas y directorios en lo que se refiere a venta, intercambio o alquiler de los datos que les suministra. Solicite que sus datos personales no vayan unidos a su identificación de acceso a Internet.

Finalidad para la que se recogen los datos

Desconfíe si los datos que le solicitan son excesivos para la finalidad con la que se recogen o innecesarios para el servicio que se le presta. Tenga en cuenta que cuando introduce su dirección de correo electrónico en un directorio, lista de distribución o grupo de noticias, dicha dirección puede ser recogida por terceros para ser utilizada con una finalidad diferente, como por ejemplo, remitirle publicidad no deseada.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

Cuando navegue por Internet, sea consciente de que los servidores Web que visita pueden registrar tanto las páginas a las que accede como la frecuencia y los temas o materias por las que busca, aunque no le informen de ello.

Asimismo, su pertenencia a determinados grupos de noticias y listas de distribución puede contribuir a la elaboración de perfiles más o menos detallados sobre su persona.

En el caso de que no desee dejar constancia de sus actividades en la red, utilice los mecanismos para preservar el anonimato que se describen en el cuerpo de este documento.

Seguridad en el intercambio de datos

Utilice, siempre que sea posible, las últimas versiones de los programas navegadores, ya que cada vez suelen incorporar mejores medidas de seguridad. Considere la posibilidad de activar en dichos programas las opciones que alerten sobre los intercambios de datos no deseados y no rellene aquellos datos que no desee hacer públicos (por ejemplo, dirección de correo electrónico, nombre, apellidos, etc.).

No realice transacciones comerciales electrónicas a través de proveedores con sistemas inseguros o no fiables. Consulte el manual de su navegador para averiguar cómo informa de que se ha establecido una conexión con un servidor seguro.

Recuerde que existen sistemas de dinero electrónico que preservan el anonimato de sus compras en Internet.

Utilice los mecanismos de seguridad que tenga a su alcance para proteger sus datos de accesos no deseados. El medio más fiable para conseguirlo es el cifrado de los mismos.

Salvo que se utilicen mecanismos de integridad, autenticación y certificación (firma digital, notarios electrónicos, etc.) no confíe ciegamente en que la persona u organización que le remite un mensaje es quien dice ser y en que el contenido del mismo no se ha modificado, aunque esto sea así en la inmensa mayoría de las ocasiones.

Y para terminar:

Siempre que se le soliciten datos personales que no esté obligado legalmente a suministrar, sopesese los beneficios que va a recibir de la organización que los recoge frente a los posibles riesgos de utilización irregular de los mismos.

Ante cualquier duda sobre la legalidad de la utilización de sus datos de carácter personal, póngase en contacto con la Agencia de Protección de Datos.

La protección jurídica de programas de ordenador. Piratería informática

El Real Decreto Legislativo 1/1996, por el que se aprueba el Texto Refundido sobre Propiedad Intelectual, la protección jurídica de los programas de ordenador, antes regulada por la Ley de Protección Jurídica de Programas de Ordenador y por la Ley de Propiedad Intelectual, crea un marco jurídico en contra de la piratería informática.

El Texto Refundido desarrolla una serie de medidas para combatir la piratería informática, como la posibilidad de que los fabricantes de programas de ordenador soliciten a la justicia española la realización de un registro sorpresa en empresas en las que existan sospechas fundadas o evidencias de delito.

España es uno de los países en los que se puede acudir a esta medida cautelar. De esta manera se erradica la posibilidad de que los presuntos infractores puedan destruir las pruebas existentes, lo cual, indudablemente ocurrirá si se les notifica por adelantado la realización de un registro.

¿En qué casos se infringe la Ley?

Al copiar o distribuir un programa de ordenador o la documentación que le acompaña, incluidas aplicaciones, datos, códigos y manuales, sin permiso expreso o licencia del propietario de los derechos de explotación.

Al utilizar un programa sin la correspondiente licencia o autorización del fabricante, con independencia de que se utilice en un solo ordenador o en varios de forma simultánea.

Al utilizar programas de ordenador en un número de copias superior al autorizado por el fabricante en sus contratos o licencias de uso.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

En empresas y demás organizaciones, al fomentar, consciente o inconscientemente, permitir, obligar o presionar a los empleados a realizar o distribuir copias no autorizadas del programa.

Al efectuar copias no autorizadas porque alguien lo requiere u obliga a ello. Al ceder o prestar el programa de forma que pueda ser copiado o al copiarlo mientras está en su posesión en calidad de cedido o prestado.

Al crear, importar, poseer o negociar con artículos destinados a burlar o neutralizar cualquier medio técnico aplicado para proteger el programa de ordenador.

Medidas Judiciales

Si finalmente existe evidencia de delito, las medidas judiciales que pueden adoptarse son:

- Solicitar al Juez un registro sorpresa de las instalaciones del presunto infractor, tanto por la vía civil, como por la penal.
- Solicitar al Juez la adopción urgente de medidas cautelares de protección.
- Exigir indemnizaciones acordes con los daños materiales y morales causados.
- El cierre del centro de actividad del infractor.
- El secuestro de todos aquellos medios destinados a suprimir los dispositivos técnicos que protegen un programa desarrollado y comercializado por un fabricante de programas.

Menores y tratamientos de datos personales

La protección de los datos personales de los menores adquiere una especial relevancia por la confluencia de distintos factores:

- Las condiciones subjetivas del titular de los datos en cuanto a su capacidad de entender y discernir sobre su derecho fundamental.
- El alto valor económico de su información personal para determinados sectores del mercado.
- La conciliación de los deberes parentales de control con los derechos del menor: vigilancia escolar, ocio y relaciones
- El uso instrumental del menor como fuente de información sobre su entorno.

Perfiles y usos

- Tratamiento de datos personales de menores:

- El menor estudiante.
 - Gestión de expedientes.
 - Controles de asistencia.
 - Responsabilidades paternas y maternas.
- El menor consumidor.
 - Objeto de productos específicos: SMS
 - Perfiles y publicidad.
- El menor y el ocio.
 - Videojuegos.
- El menor internauta.
 - Chat (foros)
 - IRC (messenger, yahoo, google...)
 - Juegos online
- El menor en el mercado de trabajo: autorizaciones para la contratación.
- El menor controlado.
 - Filtros de Internet.
 - Geolocalización

- El menor y la familia: fuente de información sobre el grupo familiar.

Problemas

- La identificación del menor:
 - ¿Es quien dice ser?
 - ¿Tiene esa edad?...
- Capacidad legal para consentir: prueba de la edad y madurez.
- La inteligibilidad de la información en la recogida de datos.
- Proporcionalidad en el tratamiento: el menor como instrumento.
 - ¿En qué trabaja tu padre?
 - ¿Qué marca de coche tenéis...?

¿Cultura de protección de datos?

- ¿Conocen los menores el derecho fundamental a la protección de datos?
 - No existe razón alguna para que tengan mayor conocimiento que el resto de la población.
- ¿Reciben algún tipo de formación escolar?
 - Ausencia de acciones específicas: el ejemplo de la educación para la salud o la nutrición no siempre se traslada al mundo internet.
 - El potencial de las asignaturas ordinarias: ética, ciencias sociales, educación para la ciudadanía.
- ¿Son capaces de discernir conceptos como “tratamiento” a “consentimiento”?

Aportaciones de la Agencia Española de Protección de Datos

- Informes jurídicos. Consentimientos :

En consecuencia, a tenor de las normas referidas, cabe considerar que los mayores de catorce años disponen de las condiciones de madurez precisas para consentir, por sí mismo, el tratamiento automatizado de sus datos de carácter personal.

Respecto de los restantes menores de edad, no puede ofrecerse una solución claramente favorable a la posibilidad de que por los mismos pueda prestarse el consentimiento al tratamiento, por lo que la referencia deberá buscarse en el artículo 162 1o del Código Civil, tomando en cuenta, fundamentalmente, sus condiciones de madurez.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

En consecuencia, a la vista de lo anteriormente señalado, será necesario recabar el consentimiento de los menores para la recogida de sus datos, con expresa información de la totalidad de los extremos contenidos en el artículo 5.1 de la Ley, recabándose, en caso de menores de catorce años cuyas condiciones de madurez no garanticen la plena comprensión por los mismos del consentimiento prestado, el consentimiento de sus representantes legales.

PROYECTO DE REAL DECRETO POR EL QUE SE APRUEBA EL REGLAMENTO DE DESARROLLO DE LA LEY ORGÁNICA 15/1999, DE 13 DE DICIEMBRE DE PROTECCIÓN DE DATOS DE CARÁCTER PERSONAL.

Artículo 13. Consentimiento para el tratamiento de datos de menores de edad.

- Acciones promocionales: derechos de los menores en Internet.
- Plan sectorial

Capacidad

- Pueden consentir los mayores de catorce años.
- Necesitan la asistencia de los titulares de la patria potestad o tutela:
 - Los mayores de catorce años cuando la Ley la exija.
 - En el caso de los menores de catorce años se requerirá el consentimiento de los padres o tutores.

Limitaciones: la información del grupo familiar

- Prohibición de utilizar al menor para recabar datos que permitan obtener información sobre los demás miembros del grupo familiar, o sobre las características del mismo sin el consentimiento de los titulares de tales datos :
 - actividad profesional de los progenitores,
 - información económica,
 - datos sociológicos
 - cualesquiera otros
- Podrán recabarse los datos de identidad y dirección del padre, madre o tutor con la única finalidad de recabar la autorización para el tratamiento de los datos de los menores.

Información en la recogida

- Cuando el tratamiento se refiera a datos de menores de edad, la información dirigida a los mismos deberá expresarse en un lenguaje que sea fácilmente comprensible por aquéllos, con expresa indicación de lo dispuesto en este artículo.

Comprobación de la edad

- Corresponderá al responsable del fichero o tratamiento articular los procedimientos que garanticen que se ha comprobado de modo efectivo la edad del menor y la autenticidad del consentimiento prestado en su caso, por los padres, tutores o representantes legales.

Conclusiones

- Apuesta clara por la tutela de los menores.
- Protección del “grupo familiar”.
- Regulación basada en la fijación de procedimientos:
 - Consentimiento
 - Información
 - Verificación de la edad.

Aportaciones

El presente Trabajo de Fin de Título trata de dar respuesta mediante el uso de las nuevas tecnologías móviles y sus aplicaciones en nuestra sociedad, a la enseñanza de las matemáticas.

Esta enseñanza de las matemáticas está orientada para alumnos de 5° de primaria, de forma que tengan un aprendizaje sobre la materia de manera autónoma y lúdica tratando de sacarle provecho a las tecnologías móviles; en nuestro caso dispositivos Tablet basados en Android.

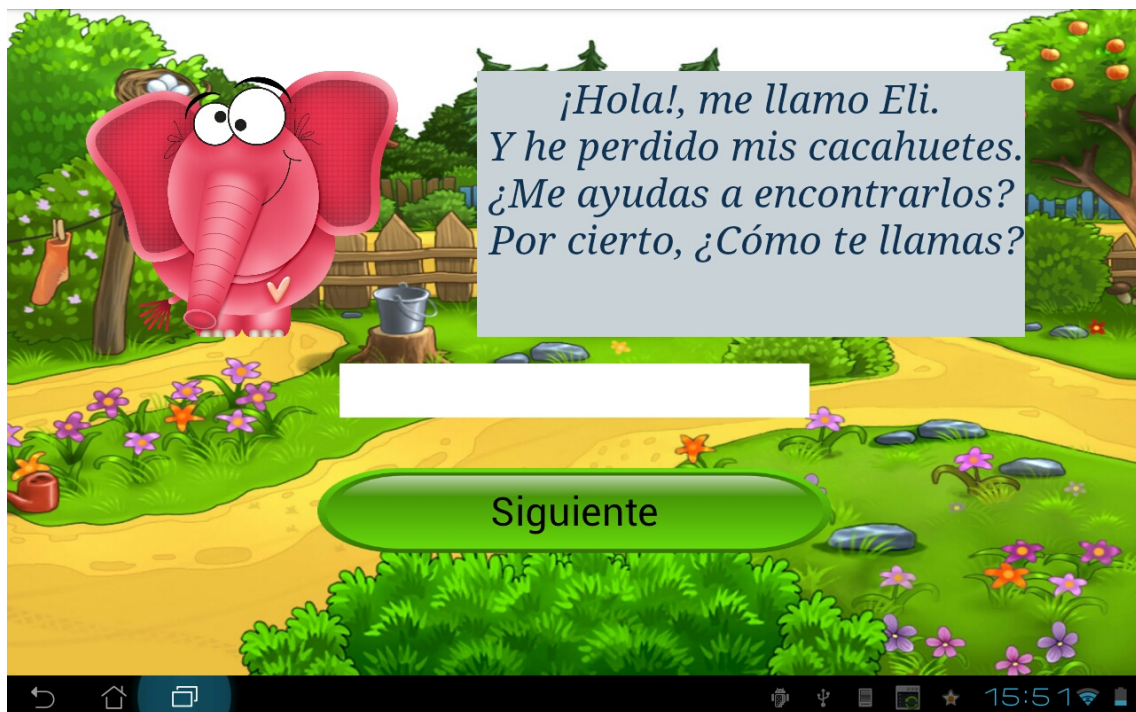
Análisis

El análisis de sistemas es la ciencia encargada del análisis de sistemas grandes y complejos, junto a la interacción entre esos sistemas. Ello engloba una de las etapas del ciclo de vida de un sistema informático, estando muy relacionado con la investigación de operaciones.

En esta etapa los analistas se encargan de analizar los requisitos del sistema, a veces se incluye en esta etapa la recolección de los requisitos del sistema. A raíz de esta etapa, se centra la atención sobre la interacción de los usuarios con el sistema.

A continuación veremos en detalle, el desarrollo de la interfaz de la aplicación y su flujo de interacción:

1-. Actividad Inicial



Esta interfaz presenta al alumno el personaje del juego, y pidiendo su nombre se crea el primer usuario activo de la aplicación. Al hacer click en el botón Siguiete se generan en la Base de Datos “ProMat5” los datos y tablas necesarias para poder hacer un uso completo y correcto de la aplicación.

2-. Menú Principal



Aquí nos encontramos ante el menú principal. En la parte superior de la pantalla como cabecera podemos ver el nombre del usuario y su puntuación total durante el juego. Esta cabecera aparecerá en diversas interfaces de la aplicación, permitiendo conocer el usuario actual activo y su puntuación.

A continuación podemos observar en la parte izquierda de la pantalla al personaje de la aplicación. Éste nos indicará un mensaje sobre él, incitando al usuario a realizar la búsqueda de sus cacahuets perdidos. Bajo el mismo, se sitúan tres cruces que representan los tres cacahuets que a perdido el personaje.

A medida que vaya avanzando el usuario y vaya superando los tres niveles se irán desbloqueando (encontrando) los cacahuets de nuestro personaje. Superado el nivel 1, aparecerá bajo el personaje un cacahuete y dos cruces (Ver Figura 1). Superado el nivel 2, aparecerán bajo el personaje dos cacahuete y una cruz (Ver Figura 2). Y superado el nivel 3, aparecerán bajo el personaje los tres cacahuets perdidos (Ver Figura 3), dando por finalizada la búsqueda de los cacahuets.

Fuera de la figuración del personaje y la búsqueda de los cacahuetes se oculta la verdadera razón de la aplicación. Al ir avanzando por los niveles, el usuario habrá ido resolviendo problemas matemáticos con orden de dificultad creciente. La inclusión del personaje y la búsqueda de los cacahuetes, entra dentro de la adaptación didáctica y lúdica de la aplicación para sus futuros usuarios; alumnos de 5º primaria.

-. Figura 1



-. Figura 2



-. Figura 3



2.1 -. Menú Principal – Salir

Al pulsar este botón conseguimos que se cierre la aplicación. Aunque todavía estará disponible en memoria, puesto que, sólo el sistema operativo tiene la potestad para gestionar la memoria del dispositivo.

Para cerrar la aplicación completamente, es decir, evitar que siga ocupando memoria principal, es necesario realizarlo mediante los mecanismos que ofrece el sistema operativo Android; específicos para ese propósito.

2.2 Menú Principal – Email

Pulsando este botón, tendremos acceso a una vista en la que se puede rellenar un formulario con el fin de añadir un problema, a modo de feedback entre la aplicación y el usuario.

Este feedback se realizará a través del envío del formulario a través de email. De tal forma, que tras rellenar el formulario (Ver figura 4) al darle a enviar, podamos enviar a través del sistema de correo electrónico Gmail dicha información. (Ver figura 5).

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

El envío de esta información será recibida en la dirección de correo creada al efecto : tfgpromat5@gmail.com

ProMatTFG

Formulario De Envio De Problemas

Enunciado	Inserte Enunciado
Solución Correcta	Inserte Solucion Correcta
Solución Errónea 1	Inserte Solucion Errónea 1
Solución Errónea 2	Inserte Solucion Errónea 2
Operación Solución Correcta	Inserte Operación Solución Correcta
Nivel Problema (facil-1 ; medio-2; dificil-3)	Inserte Nivel del Problema

Volver Enviar

15:56

-. Figura 4

ProMatTFG

Formulario De Envio De Problemas

Enunciado	¿Cuánto es 2 más 2?
Solución Correcta	4
Solución Errónea 1	22
Solución Errónea 2	3
Operación Solución Correcta	2+2=4
Nivel Problema (facil-1 ; medio-2; dificil-3)	1

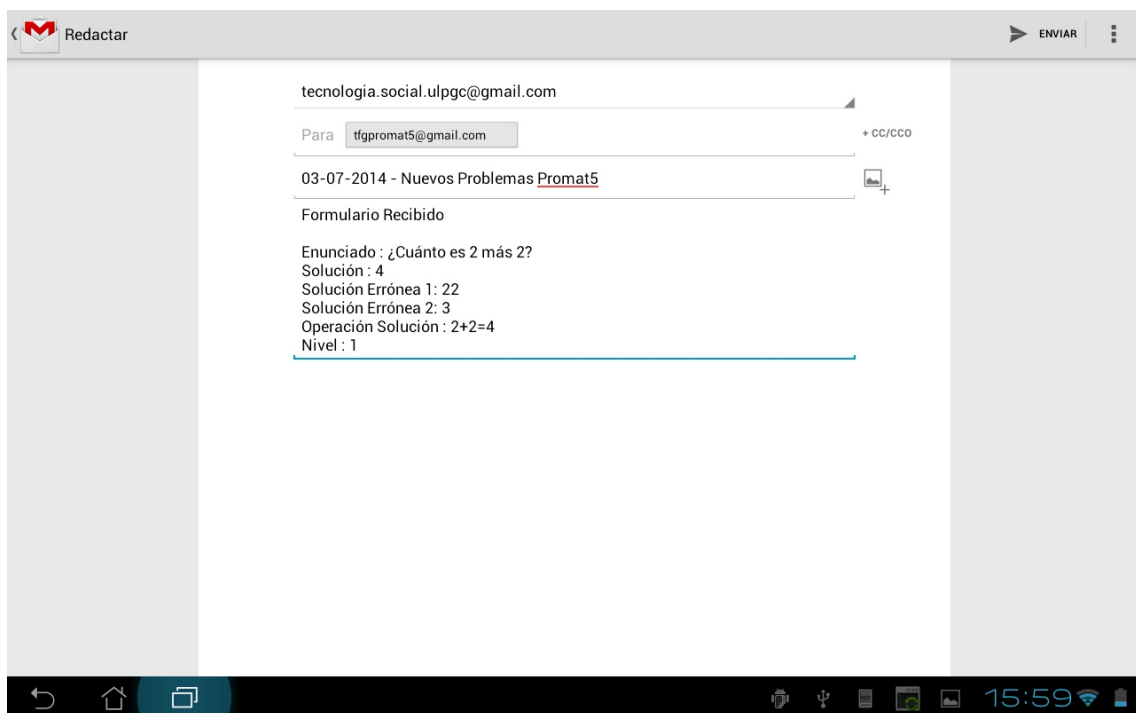
Volver Enviar

15:58

-. Figura 5



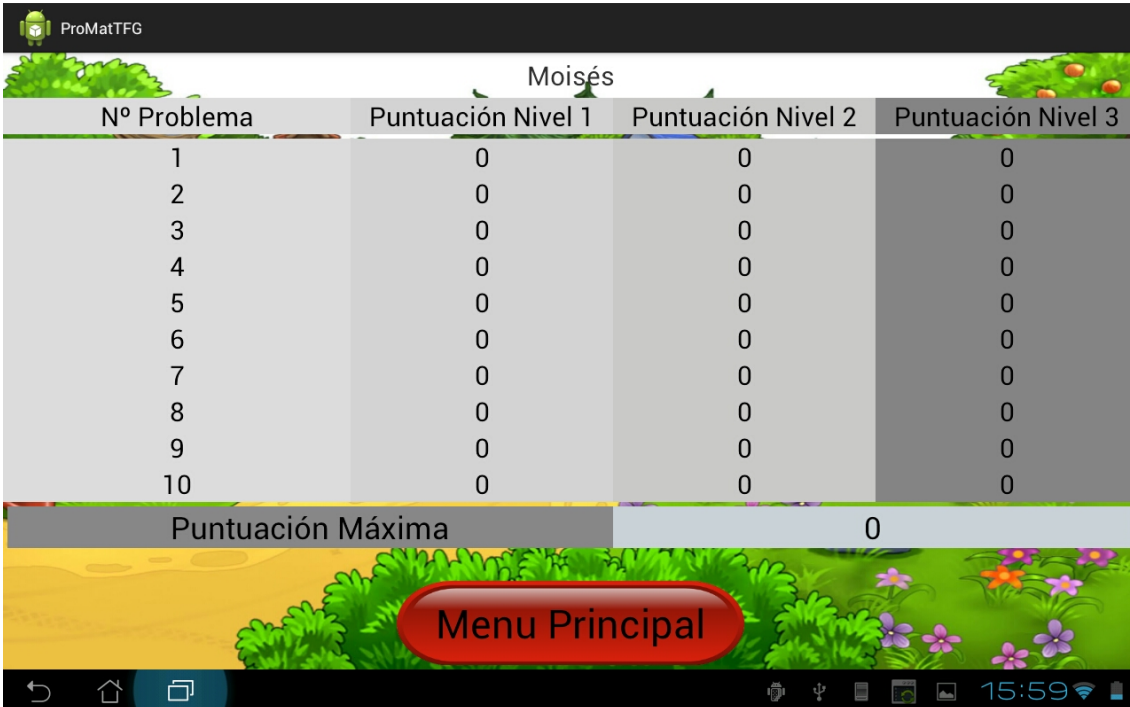
Tras tener todo el formulario relleno y preparado para ser enviado a través del gestor de correo electrónico Gmail a la dirección de correo anteriormente mencionada. Esta información es tratada de la siguiente manera:



2.3 Menú Principal – Puntos

A través de este botón, podemos ver en una interfaz tabulada, la información acerca de las puntuaciones del usuario activo actual para todos los problemas y niveles; Junto a la puntuación total que ha obtenido el usuario activo actual hasta el momento.

A continuación se observa el estado inicial de la información tabulada para el usuario; el nombre del mismo aparece en la parte superior de la pantalla:



Moisés			
Nº Problema	Puntuación Nivel 1	Puntuación Nivel 2	Puntuación Nivel 3
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	0	0	0
6	0	0	0
7	0	0	0
8	0	0	0
9	0	0	0
10	0	0	0
Puntuación Máxima		0	

Menu Principal

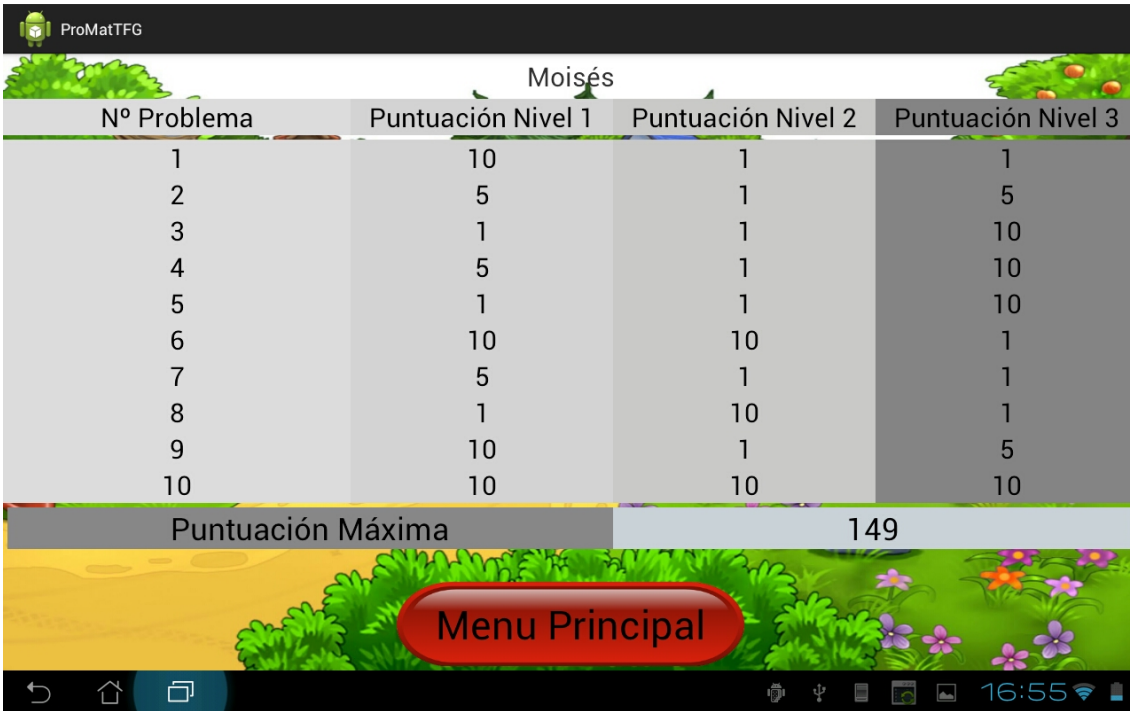
Ejemplo de puntuaciones para el nivel 1:

Moisés			
Nº Problema	Puntuación Nivel 1	Puntuación Nivel 2	Puntuación Nivel 3
1	10	0	0
2	5	0	0
3	1	0	0
4	5	0	0
5	1	0	0
6	10	0	0
7	5	0	0
8	1	0	0
9	10	0	0
10	10	0	0
Puntuación Máxima		58	

Ejemplo de puntuaciones para el nivel 2:

Moisés			
Nº Problema	Puntuación Nivel 1	Puntuación Nivel 2	Puntuación Nivel 3
1	10	1	0
2	5	1	0
3	1	1	0
4	5	1	0
5	1	1	0
6	10	10	0
7	5	1	0
8	1	10	0
9	10	1	0
10	10	10	0
Puntuación Máxima		95	

Ejemplo de puntuaciones para el nivel 3:



Moisés			
Nº Problema	Puntuación Nivel 1	Puntuación Nivel 2	Puntuación Nivel 3
1	10	1	1
2	5	1	5
3	1	1	10
4	5	1	10
5	1	1	10
6	10	10	1
7	5	1	1
8	1	10	1
9	10	1	5
10	10	10	10
Puntuación Máxima		149	

Menu Principal

En todos los casos podemos apreciar que en la parte inferior de la pantalla se encuentra el botón Menú Principal. Que nos permite como su título indica, volver hacia la vista del Menú Principal.

2.4 -. Menú Principal – Usuarios

Gracias al botón Usuarios podemos administrar los perfiles de los usuarios de la aplicación. La aplicación está diseñada de manera que puedan haber como máximo 4 usuarios independientes entre sí que permiten utilizar la aplicación. Como mínimo, siempre habrá un usuario.

Ese usuario será el usuario activo actual, que será a partir del cual la aplicación maneje y administre su información, independientemente de si hay o no más usuarios.

En la vista que nos ofrece la administración de los perfiles de los usuarios de la aplicación, en la parte izquierda de la pantalla podemos observar los usuarios disponibles; teniendo marcado al usuario actual mediante un radioButton activo. Y en la parte derecha de la misma, podemos ver un listado de botones con las posibles acciones que se permiten realizar sobre los usuarios de la aplicación.



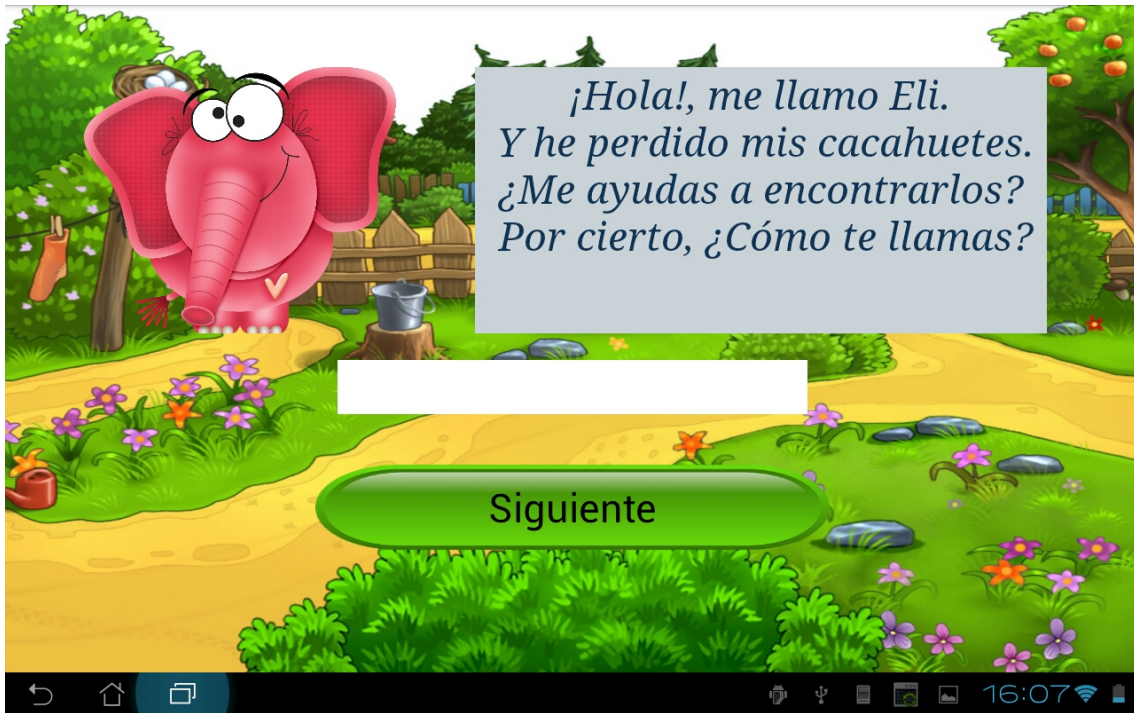
En este apartado de la administración de perfiles de usuario de nuestra aplicación, son posibles de realizar las siguientes acciones:

2.4.1 Menú Principal – Usuarios – Añadir Usuario

Esta acción nos permite siempre que haya algún usuario libre (un usuario libre es denotado por la terminación “Nouser”) en el listado de usuarios. Si es así, se creará el usuario en la base de datos y estará disponible para hacer uso de la aplicación. En caso contrario, no será posible la creación del nuevo usuario.

Para poder hacer uso de la aplicación, este nuevo usuario deberá ser seleccionado previamente mediante la acción “Cambiar Usuario” que será descrita a posteriori.

Al crear un nuevo usuario, se insertará su nombre a través de la siguiente interfaz:



2.4.2 Menú Principal – Usuarios – Editar Usuario

Esta acción nos permite modificar el nombre de un usuario existente en la aplicación a través de la siguiente interfaz:



2.4.3 Menú Principal – Usuarios – Cambiar Usuario

Esta acción nos permite cambiar el usuario activo actual. Seleccionando en la parte izquierda de la pantalla un usuario diferente de Nouser (Usuarios Inactivos) y posteriormente pulsando el botón “Cambiar Usuario”, cambiamos el estado del usuario seleccionado a ser el nuevo usuario activo actual de nuestra aplicación.

2.4.4 Menú Principal – Usuarios – Eliminar Usuario

Esta acción nos permite eliminar un usuario y todo su contenido de nuestra aplicación. La única excepción de esta acción es que no puede ser eliminado el usuario activo actual.






2.5 -. Menú Principal – Jugar

A partir de este botón es donde comienza el peso de la aplicación. Trataré de explicar de la forma más idónea el flujo de la aplicación. Nada más accionar el botón de “Jugar” nos encontramos ante la vista de la selección de niveles.



Esta vista tiene en la parte superior de la pantalla la cabecera con los datos (nombre y puntuación) del usuario actual activo. Luego, en la parte central de la pantalla, encontramos los botones de acceso a los niveles junto a una imagen indicativa al estado actual del nivel para el usuario actual activo. Los niveles tienen 3 tipos de estado:

Estado	Imagen Representativa
Desbloqueado	
Bloqueado	
Finalizado	

- Ejemplo Nivel 1 Finalizado



- Ejemplo Nivel 2 Finalizado



Solamente se tendrá acceso a un nivel cuando el estado del mismo sea desbloqueado, y a partir de ella se irá a la vista que contiene los problemas que se pueden realizar. En caso contrario, no permitirá el acceso al nivel y mostrará un mensaje mediante un diálogo tipo Toast.

En la parte inferior de esta vista, encontraremos un botón de retorno al menú principal.

2.5.1 -. Menú Principal –Jugar - Listado Problemas

Tras acceder a un nivel cuyo estado es desbloqueado, accedemos a esta vista con el listado de problemas de dicho nivel junto a una imagen representativa del estado del problema.

Dichas imágenes representativas de los estados de los problemas serán las mismas y con el mismo significado que para los niveles anteriormente vistos y explicados.



- Ejemplo vista problemas



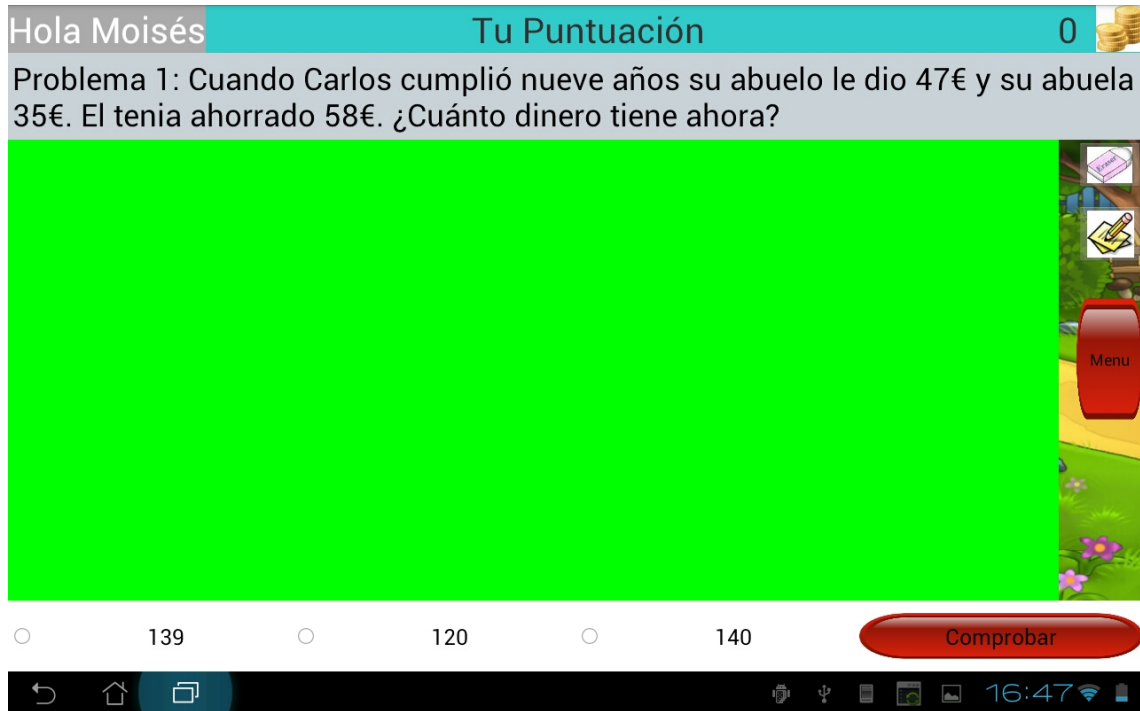
Esta vista tiene en la parte superior de la pantalla la cabecera con los datos (nombre y puntuación) del usuario actual activo. Luego, en la parte central de la pantalla, encontramos los botones de acceso a los problemas junto a una imagen indicativa al estado actual del problema para el usuario actual activo.

Solamente se tendrá acceso a un problema cuando el estado del mismo sea desbloqueado, y a partir de ella se irá a la vista que contiene la estructura de realización del problema que se puede realizar (Será visto en el siguiente punto). En caso contrario, no se permitirá el acceso al problema y mostrará un mensaje mediante un diálogo tipo Toast.

En la parte inferior de esta vista, encontraremos un botón de retorno al menú principal.

2.5.2 -. Menú Principal –Jugar – Realización de Problemas

Al llegar a esta vista nos encontramos con la parte donde el usuario de la aplicación va a obtener contacto para adquirir el conocimiento en la materia de las matemáticas. Aquí se realizarán los problemas matemáticos y esta es su vista:



Como se puede apreciar en la imagen superior, se sigue teniendo en la parte superior de la pantalla la cabecera de datos del usuario (nombre y puntuación). A continuación podemos apreciar una zona donde se situarán los diferentes enunciados de los problemas que irá realizando el usuario.

Por otro lado, en la parte central de la pantalla observamos un rectángulo verde. Este rectángulo, es un área destinada para la resolución de los problemas planteados en la aplicación, y tendrá un uso de pizarra. En la parte derecha de nuestra “pizarra” tenemos 3 botones:

El primer botón (botón con una imagen de goma) permite el borrado de la pizarra al deslizar el dedo sobre la misma.

El siguiente botón (botón con una imagen de lápiz) permite la escritura en la pizarra al deslizar el dedo sobre la misma, el color de escritura seleccionado es el blanco, tratando de imitar una pizarra escolar.

Y por último, tenemos el botón “Menú”, el cual nos permite volver al menú principal.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

En la parte inferior de la vista tenemos un grupo de botones con las 3 posibles soluciones a los problemas. Estas soluciones son colocadas de forma aleatoria, de modo que el alumno no caiga en la resolución repetitiva de los problemas (“Y no haga trampas”).

La lógica para la realización de los problemas es la siguiente: El usuario dispone de dos intentos para solucionar de forma correcta cada problema planteado. Su puntuación y procedimiento de actuación, se irán conociendo más adelante.

El botón más situado a la derecha, es el botón “Comprobar”. Al seleccionar este botón se procederá a comprobar si la respuesta seleccionada es la correcta. El proceso para comprobar una respuesta es el siguiente:

1. Seleccionar una respuesta
2. Pulsar el botón “Comprobar”

Si la respuesta es incorrecta (es el primer fallo), aparecerá un mensaje de diálogo Toast indicándolo. Y se permite realizar otra selección de respuesta. Si tras volver a seleccionar una respuesta, esta resulta incorrecta de nuevo, aparecerá un mensaje de diálogo de tipo Toast indicándolo, y pasará a la siguiente interfaz donde se representa la puntuación asignada al usuario por resolver el problema según el intento en el que se resolvió el problema actual.

Puntuaciones según el intento en el que se resolvió el problema:

- Intento 1 = Se le asignan al usuario 10 puntos. (Ver Figura 6)
- Intento 2 = Se le asignan al usuario 5 puntos. (Ver Figura 7)
- Intento 3 = Se le asigna al usuario 1 punto. (Ver figura 8)

- Ejemplo de uso de esta interfaz

Hola Moisés Tu Puntuación 0

Problema 1: Cuando Carlos cumplió nueve años su abuelo le dio 47€ y su abuela 35€. El tenía ahorrado 58€. ¿Cuánto dinero tiene ahora?

Handwritten calculations on a green background:

$$\begin{array}{r} 47 \\ + 35 \\ \hline 82 \text{€} \end{array}$$
$$\begin{array}{r} 82 \\ + 58 \\ \hline 140 \text{€} \end{array}$$

139 120 140 Comprobar

16:48

- Figura 6

Excelente!!!!
Encontraremos mis cacahuetes!!!!
Muy rápido si sigues así !!!

Has obtenido una Puntuación de 10 PUNTOS!!!

Siguiente

16:48

-Figura 7



-Figura 8



Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

Tras la realización completa de los problemas de cada nivel se irán desbloqueando los niveles siguientes y de forma paralela se irán consiguiendo los cacahuetes de nuestro personaje.

- Nivel 1 desbloqueado



- Nivel 2 desbloqueado



- Nivel 3 desbloqueado



Si el nivel superado es el nivel 3, se da por finalizado el juego por completo. Y muestra la siguiente vista:



Tras finalizar todos los problemas, ya no se pueden volver a realizar ningún otro problema de ningún nivel. Sólo se podrá ver la puntuación del usuario.

Requisitos Hardware y Software

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

Los medios necesarios para la elaboración del trabajo requieren la utilización de hardware, software u otro material especializado que no sea de uso general se deberá explicitar la manera en que se va a tener acceso a dichos medios.

Los recursos necesarios que se han utilizado para realizar este trabajo son:

Microsoft Word

El Microsoft Word se ha empleado fundamentalmente como procesador de los documentos relacionados con la creación de la memoria correspondiente al trabajo de fin de grado.

StartUML

Se ha empleado esta aplicación para reflejar de una manera gráfica los diferentes aspectos de la aplicación que se ha desarrollado. StartUML se utiliza bajo licencia GNU GPL.

Android Development Tools (ADT)

Android Development Tools (ADT) es un plugin para el IDE Eclipse que está diseñado para darle un poderoso entorno integrado en el que la construcción de aplicaciones de Android.

ADT amplía las capacidades de Eclipse para permitirle configurar rápidamente nuevos proyectos para Android, crear una interfaz de usuario de la aplicación, agregue los paquetes basados en la API de Android Framework, depurar sus aplicaciones utilizando las herramientas del SDK de Android, e incluso exportar firmado (o signo) .apk archivos con el fin de distribuir la aplicación.

El desarrollo en Eclipse con ADT es muy recomendable y es la manera más rápida de comenzar. Con la configuración del proyecto guiada que ofrece, así como la integración de herramientas, editores de XML personalizados, y el panel de resultados de la depuración, ADT le da un impulso increíble en el desarrollo de aplicaciones Android.

SQLite

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

SQLite es un motor de bases de datos muy popular en la actualidad por ofrecer características tan interesantes como su pequeño tamaño, no necesitar servidor, precisar poca configuración, ser transaccional y por supuesto ser de código libre.

Android incorpora de serie todas las herramientas necesarias para la creación y gestión de bases de datos SQLite, y entre ellas una completa API para llevar a cabo de manera sencilla todas las tareas necesarias.

A diferencia de los sistemas de gestión de bases de datos cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo.

El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos.

El conjunto de la base de datos (definiciones, tablas, índices, y los propios datos), son guardados como un sólo fichero estándar en la máquina host. Este diseño simple se logra bloqueando todo el fichero de base de datos al principio de cada transacción.

SQLiteStudio

SQLiteStudio es un administrador de base de datos SQLite que nos permite importar nuestra base de datos del dispositivo al ordenador para comprobar su contenido o realizar consultas sobre ella.

Desarrollo

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

La metodología empleada en este trabajo responde a una orientación evolutiva, permitiendo seguir secuencias ascendentes o descendentes en las etapas del desarrollo.

Dicha metodología reconoce las prioridades de los usuarios para así partir de múltiples versiones, donde cada una incorpora la experiencia de las anteriores, así el desarrollo de la siguiente versión no comienza de inmediato, sino que necesita experimentar con cada versión antes de comenzar con la siguiente para refinar los requisitos.

De esta manera, una vez finalizada una versión se utiliza en paralelo con la anterior hasta que la última sea finalmente aceptada, sin implicar que cada requisito sea implementado en cada ciclo.

El modelo evolutivo se diferencia del modelo por prototipos en que se da por hecho que aunque se necesiten varias iteraciones para lograrlo al final se llegará a tener una serie de requisitos completos y sin errores, que no vayan a cambiar más. En el modelo evolutivo se asume que los requisitos pueden cambiar en cualquier momento del ciclo de vida y no solo en la etapa de análisis.

En el proceso evolutivo y por tanto en el desarrollo del presente trabajo se llevan a cabo las siguientes fases:

DES= RU + RS + DD + I + T

RU: Requerimiento del usuario.

RS: Requerimiento del software.

DD: Diseño detallado y producción.

I: Implementación.

T: Transferencia.

Partiendo del presente trabajo corroboramos las ventajas que resultan de esta

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

metodología, ya que la experiencia del usuario es necesaria, de acuerdo a nuevos requerimientos de usuarios y complejos dependientes de la tecnología no disponible.

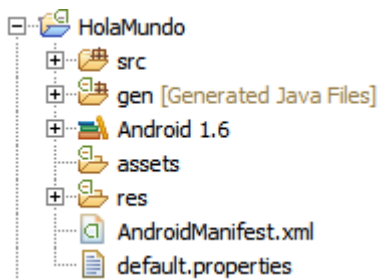
A nivel de desventajas se podría valorar las soluciones temporales empotradas en el sistema y las nuevas versiones más caras para el sistema, que sin duda siempre serán beneficiosas para la población sorda o con discapacidad auditiva, lo que de alguna manera contrarresta el gasto en las versiones para el sistema.

	RU	RS	D	I	T	Total
Iteración 1	20	20	30	60	10	140
Iteración 2	5	5	25	45	10	90
Iteración 3	5	5	20	30	10	70
Total	30	30	75	135	30	300

Antes de programar es necesario conocer la estructura de una aplicación Android ya que es importante saber qué es cada elemento.

Cuando creamos un nuevo proyecto Android en Eclipse se genera automáticamente la estructura de carpetas necesaria para poder generar posteriormente la aplicación. Esta estructura será común a cualquier aplicación, independientemente de su tamaño y complejidad.

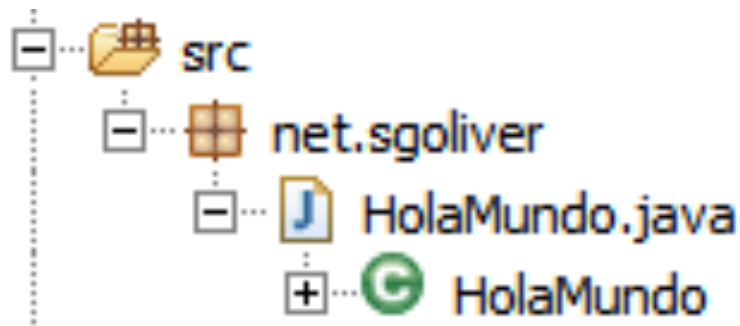
En la siguiente imagen vemos los elementos creados inicialmente para un nuevo proyecto Android:



Describamos los elementos principales:

Carpeta /src/:

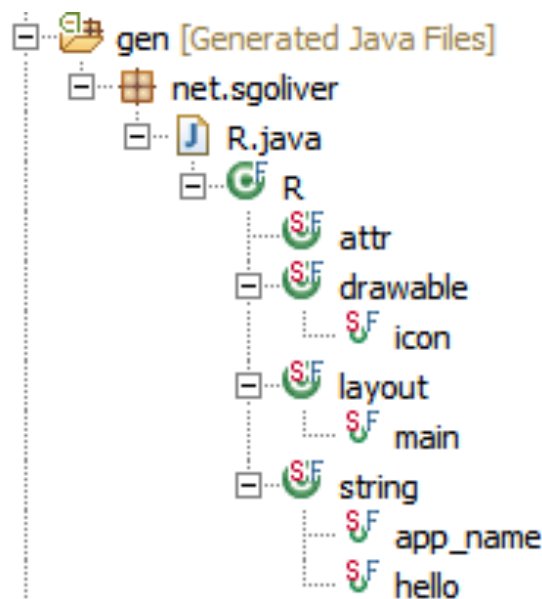
Contiene todo el código fuente de la aplicación, código de la interfaz gráfica, clases auxiliares, etc. Inicialmente, Eclipse creará por nosotros el código básico de la pantalla (Activity) principal de la aplicación, siempre bajo la estructura del paquete java definido.



Carpeta /gen/:

Contiene una serie de elementos de código generados automáticamente al compilar el proyecto ya que nosotros no podemos modificarla manualmente.

Cada vez que generamos nuestro proyecto, la maquinaria de compilación de Android genera por nosotros una serie de ficheros fuente en java dirigido al control de los recursos de la aplicación.



El más importante es el que se puede observar en la imagen, el fichero R.java, y la clase R.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

Esta clase R contendrá en todo momento una serie de constantes con los ID de todos los recursos de la aplicación incluidos en la carpeta /res/, de forma que podamos acceder fácilmente a estos recursos desde nuestro código a través de este dato.

Así, por ejemplo, la constante R.drawable.icon contendrá el ID de la imagen “icon.png” contenida en la carpeta /res/drawable/. Veamos como ejemplo la clase R creada por defecto para un

```
proyecto nuevo:
1 package net.sgoliver;
2
3 public final class R {
4     public static final class attr {
5     }
6     public static final class drawable {
7         public static final int icon=0x7f020000;
8     }
9     public static final class layout {
10        public static final int main=0x7f030000;
11    }
12    public static final class string {
13        public static final int app_name=0x7f040001;
14        public static final int hello=0x7f040000;
15    }
16 }
```

Carpeta/Android 1.6 :

Cuando al principio seleccionamos una versión de Android para nuestra aplicación, se incluye la librería correspondiente (android.jar) y nuestro proyecto hace referencia desde esta carpeta. Estas librerías dependen del SDK con el que estemos trabajando.

Carpeta /assets/:

Contiene todos los demás ficheros auxiliares necesarios para la aplicación (y que se incluirán en su propio paquete), como por ejemplo ficheros de configuración, de datos, etc.

Carpeta /res/:

Nos encontramos ahora con la carpeta/res/, esta carpeta almacena, mediante un

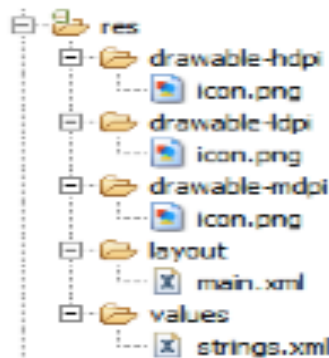
Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

sistema de subcarpetas, los recursos de la aplicación, desde imágenes hasta los elementos .xml que compondrán el interfaz de usuario y la funcionalidad del mismo.

Contiene todos los ficheros de recursos necesarios para el proyecto: imágenes, vídeos, cadenas de texto, etc. Los diferentes tipos de recursos de deberán distribuir entre las siguientes carpetas:

- /res/drawable/. Contienen las imágenes de la aplicación. Se puede dividir en /drawable-ldpi, /drawable-mdpi y /drawable-hdpi para utilizar diferentes recursos dependiendo de la resolución del dispositivo.
- /res/layout/. Contienen los ficheros de definición de las diferentes pantallas de la interfaz gráfica.
- /res/values/. Contiene otros recursos de la aplicación como por ejemplo cadenas de texto (strings.xml), estilos (styles.xml), colores (colors.xml), etc.

Como ejemplo, para un proyecto nuevo Android, se crean los siguientes recursos para la aplicación:



Fichero AndroidManifest.xml:

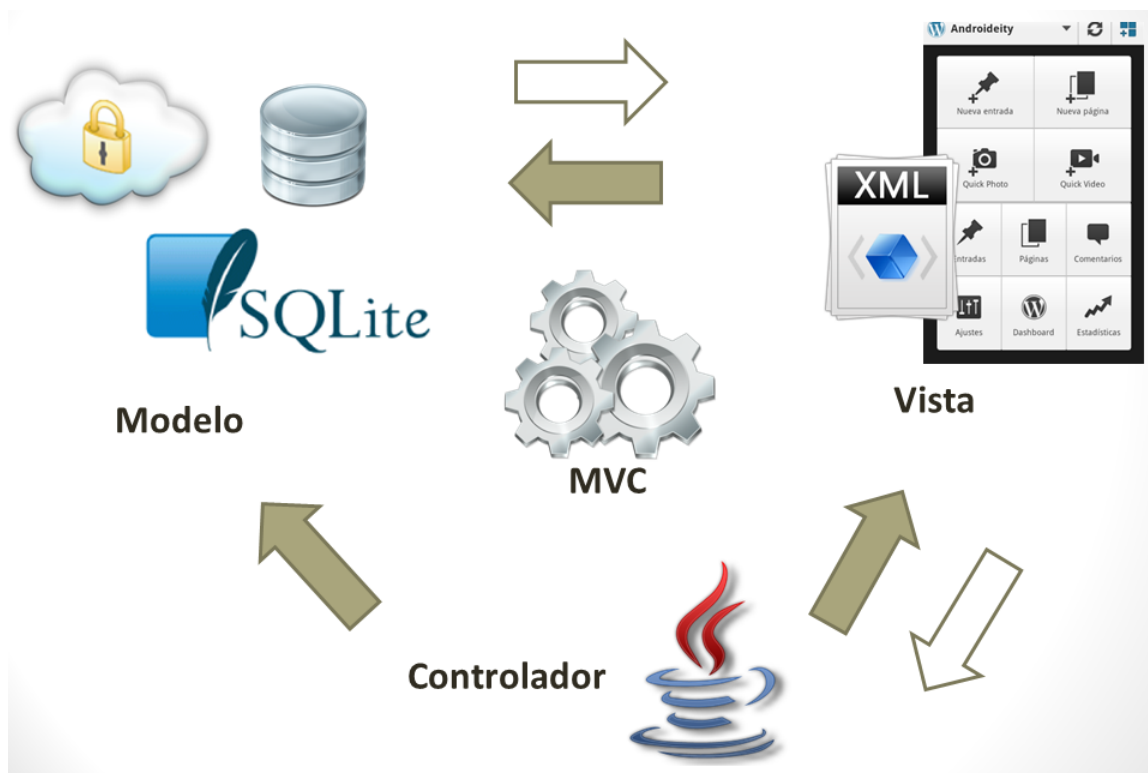
Contiene la definición en XML de los aspectos principales de la aplicación, como por ejemplo su identificación (nombre, versión, icono,...), sus componentes (pantallas, mensajes,...), o los permisos necesarios para su ejecución.

Diseño

En Android utilizamos el patrón de arquitectura llamado **Modelo Vista Controlador (MVC)** cuya principal bondad consiste en separar los datos de una aplicación, la interfaz de usuario y la lógica de negocios en tres componentes distintos que se relacionarán para al final tener como resultado nuestra aplicación.

De esta forma podemos seccionar de forma más fácil nuestro equipo de trabajo y dedicarnos a desarrollar nuestros componentes de tal forma que construyamos módulos o librerías con funcionalidades específicas que incluso podríamos reutilizar en proyectos posteriores y no simplemente en el proyecto actual. Para lograr esto, el diseño de la arquitectura de nuestra aplicación juega un papel importante y la capacidad de abstracción que tengamos desarrollada.

Ejemplo visual de los componentes de este modelo:



- **Modelo.** Nos referimos con modelo a las representaciones que construiremos basadas en la información con la que operará nuestra aplicación. En Java, el modelo viene siendo análogo a los *beans* que tienen la particularidad de ser reutilizables y nos ayudan a cumplir con el proverbio de oro **“Don’t Repeat Yourself” (DRY)** haciendo a nuestras aplicaciones escalables. En esta parte del modelo también juega la decisión de qué modelo para almacenar información utilizaré. ¿Base de datos? ¿Web Services? El modelo que elijas depende obviamente de las necesidades de información de tu aplicación.
- **Vista.** La vista no es más que la interfaz con la que va a interactuar el usuario. En Android, las interfaces las construimos en XML. Se suele utilizar mucho la analogía de que esta parte es realmente parecida a lo que hacemos en el desarrollo web con los CSS. Construimos nuestro esqueleto en XML que

equivale al HTML de un sitio. Posteriormente, con ayuda de estilos, que también los escribimos en XML, podemos empezar a darle formato de colores, posiciones, formato, etc. a nuestro esqueleto. Esto equivale a los CSS. Si se tiene un conocimiento previo sobre el estilo de desarrollo web, se puede aprovechar ese conocimiento para aplicarlo en crear aplicaciones Android.

- **Controlador.** Finalmente nos topamos con el controlador que son todas esas clases que nos ayudarán a darle vida a esas interfaces bonitas que ya construimos y nos permitirán desplegar y consumir información para el usuario. Estos controladores se programan en lenguaje Java y son el código central de la aplicación.

En el MVC podemos tener perfiles especializados en construir cada uno de estos componentes y tener un resultado de calidad en cada uno y al final obtener una buena aplicación. El truco para que todo salga bien es ser disciplinados para poder ser programadores ordenados, eso es vital en la aplicación de esta arquitectura. Por experiencia, te puedo decir que si te esfuerzas en construir una buena arquitectura de aplicación te ahorrarás muchos dolores de cabeza después y te evitarás tener que duplicar líneas de código que hacen lo mismo, podrás escalar fácilmente tu aplicación, agregar funcionalidades nuevas sin tanto embrollo y ¿por qué no?, facilitar el comienzo de nuevos proyectos.

Otra de las cosas que nos ayudan a seguir este modelo es tener ordenado nuestro proyecto. Utilizar las convenciones de los lenguajes que utilicemos y las buenas prácticas para que sea más fácil acoplar a tus compañeros de equipo (no siempre uno solo se echa el paquete de hacer una aplicación y menos aún si es una compleja). Es aconsejable empezar con cosas tan sencillas como el nombramiento de paquetes, clases y variables, y después organizar los recursos que vayamos a utilizar delegando siempre a cada componente lo que le corresponde.

En la web podemos encontrar muchos recursos que podemos leer para ver ejemplos de la aplicación de esta arquitectura. No es algo nuevo pero algunos no la conocen. Sin importar que encontremos un ejemplo para web, recuerda que en la programación únicamente cambia el lenguaje, la lógica y la metodología es la que se preserva.

Las bases de datos son una herramienta de gran potencia en la creación de aplicaciones informáticas. Hasta hace muy poco resultaba muy costoso y complejo incorporar bases de datos a nuestras aplicaciones. No obstante, Android incorpora la librería SQLite que nos permitirá utilizar bases de datos mediante el lenguaje SQL, de una forma sencilla y utilizando muy pocos recursos del sistema. Como veremos, almacenar nuestra información en una base de datos no es mucho más complejo que almacenarlos en un fichero, y además resulta mucho más potente.

SQL es el lenguaje de programación más utilizado para bases de datos. No resulta complejo entender los ejemplos sencillos que podemos observar en la web. No obstante, si deseas hacer cosas más complicadas es recomendable que consultemos

alguno de los muchos manuales que se han escrito sobre el tema, permitiendo evolucionar mucho más el desarrollo de nuestra base de datos de nuestra aplicación.

Para manipular una base de datos en Android usaremos la clase **SQLiteOpenHelper** que nos facilita tanto la creación de la base de datos, como el trabajar con futuras versiones de esta base de datos. Para crear un descendiente de esta clase hay que implementar los métodos **onCreate()**, y **onUpgrade()** y opcionalmente **onOpen()**.

La gran ventaja de utilizar esta clase es que ella se preocupará de abrir la base de datos si existe o de crearla si no existe. Incluso de actualizar la versión si decidimos crear una nueva estructura de la base de datos. Además, esta clase tiene dos métodos **getReadableDatabase()** y **getWritableDatabase()** que abren la base de datos en modo solo lectura o lectura y escritura. En caso de todavía no existir la base de datos, estos métodos se encargarán de crearla.

Diseño de Base de Datos

SQLite es un manejador open source de bases de datos que combina una interfaz muy limpia de SQL y que nos permite trabajar con poca memoria y con una velocidad bastante decente, características que son necesarias cuando hablamos de entornos móviles.

SQLite soporta las características estándar de las bases de datos relacionales como la sintaxis que se basa en SQL, transacciones y la elaboración de consultas. Debido a esto, cualquier desarrollador que haya trabajado con bases de datos sin importar el entorno, no encontrará una dificultad especial en trabajar con bases de datos locales en Android. Por otro lado, hay que dejar en claro que la API nativa de **SQLite** no es compatible con JDBC contrario a lo que podría pensarse tomando en cuenta el uso de Java como lenguaje nativo en aplicaciones Android.

Un acercamiento a SQLite

SQLite, como su nombre indica, utiliza el lenguaje SQL para las consultas (SELECT), manipulación de datos (INSERT, DELETE, etc.), y de definición de datos (CREATE TABLE, etc). **SQLite** presenta unas pequeñas variaciones donde se desvía del estándar **SQL-92**, que aplica para la mayoría de bases de datos SQL.

Entre ellas, el uso de FOREIGN KEY, transacciones anidadas, RIGHT OUTER JOIN, FULL OUTER JOIN y algunos usos de ALTER TABLE no son válidas en SQLite.

Para utilizar SQLite no se requiere de ninguna configuración o administrador de base de datos, por lo que únicamente deberemos preocuparnos por el manejo de SQL y todo lo demás se gestionará a través del entorno de desarrollo Android.

Debemos tomar en cuenta desde el principio que el uso de bases de datos implica tareas de I/O, que siempre tienen un impacto en el rendimiento de las aplicaciones, por lo que en ocasiones todo esto puede tornarse lento.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

SQLite soporta los tipos de datos TEXT (similar a los String en Java), INTEGER (similar a Integer en Java) y REAL (similar a Double en Java). Si hacemos uso de cualquier otro tipo de dato, de manera automática serán convertidos para que sean compatibles con estos tres tipos de datos.

Si nuestra aplicación crea una base de datos, esta se guarda en el directorio *DATA/data/APP_NAME/databases/FILENAME*. En nuestro caso:

“data/data/ProMatTFG/databases/ProMat5”

“DATA” es la ruta que retorna el método *Environment.getDataDirectory()*, “APP_NAME” es el nombre de nuestra aplicación, y “FILENAME” es el nombre que le damos a la base de datos al crearla. Las bases de datos en SQLite tienen un nivel de acceso privado que abarca únicamente la aplicación en la que fueron creadas.

Desglose de las Tablas de nuestra base de datos

Nombre Base de Datos : **ProMat5**

1-. Tabla Usuarios:

En esta tabla guardaremos los siguientes datos:

- Idusuario (Integer) : Código del usuario.
- Nombre (String) : Nombre del usuario.
- Juega (Integer) : Estado del usuario.
 - . Juega = 1 : Usuario Actual Activo
 - . Juega = 2 : Usuario No Activo
- Nivel (Integer) : Nivel del problema por donde va el usuario
- Idproblema (Integer) : Problema del nivel por donde va el usuario.

Esta tabla se rellena al comenzar la aplicación la primera vez. Si no es la primera vez, sólo se accede a su información de la manera oportuna. Estado de la tabla tras ser creada la primera vez que se accede a nuestra aplicación.

Suponiendo que el nombre insertado a sido “Usuario1”. El nombre “Nouser” indica que no hay usuario, y la aplicación lo detecta.

Usuarios				
Idusuario	Nombre	Juega	Nivel	Idproblema
1	Usuario1	1	1	1
2	Nouser	2	1	1
3	Nouser	2	1	1
4	Nouser	2	1	1

2-. Tabla ProblemasNivel[1|2|3]:

En esta tabla guardaremos los siguientes datos:

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

- Idproblema (Integer) : Código del problema del nivel citado.
- Enunciado (String) : Enunciado del problema.
- Solucionbuena (String) : Solución numérica del problema.
- Solucionmala1 (String) : Solución errónea 1 numérica del problema.
- Solucionmala2 (String) : Solución errónea 2 numérica del problema.
- Operación (String) : Operación numérica que resuelve el problema.

En esta tabla se guarda la información necesaria de los diferentes problemas. Habrá tres tablas para guardar dicha información dependiendo del nivel de los problemas. Para los problemas del nivel 1 el nombre de la tabla será **ProblemasNivel1**; Para los problemas del nivel 2 el nombre de la tabla será **ProblemasNivel2**; y para los problemas del nivel 3 el nombre de la tabla será **ProblemasNivel3**.

Al no variar el contenido de estas tablas durante el uso de la aplicación estas tablas son creadas en el momento de abrir la aplicación por primera vez. Continuando inalteradas durante el tiempo que nuestra aplicación permanezca en nuestro dispositivo.

Cada una de las tablas contendrá 10 filas, representando 10 problemas por nivel. A continuación, un ejemplo de una de las filas de las tablas:

ProblemasNivel1	
Idproblema	1
Enunciado	“Cuando Carlos cumplió nueve años su abuelo le dio 47€ y su abuela 35€. El tenía ahorrado 58€. ¿Cuánto dinero tiene ahora?”
Solucionbuena	“140”
Solucionmala1	“139”
Solucionmala2	“120”
Operacion	“47+35+59=140”

3-. Tabla PuntuacionesProblemasNivel[1|2|3]Usuario[1|2|3|4]:

En esta tabla guardaremos los siguientes datos:

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

- Idproblema (Integer) : Código del problema del nivel citado.
- Puntuación (Integer) : Puntuación del problema del nivel citado.
- Estado (Integer) : Estado del problema para el usuario.
 - Estado = 1. (Problema actual accesible).
 - Estado = 2. (Problema realizado no accesible).
 - Estado = 0. (Problema no realizado no accesible).

En esta tabla se guarda la información necesaria; en este caso las puntuaciones obtenidas en los problemas de los diferentes niveles y para cada usuario disponible o no. Habrá tres tablas por cada usuario para guardar dicha información dependiendo del nivel de los problemas.

Para los problemas del nivel 1 y el usuario 1 el nombre de la tabla será **PuntuacionesProblemasNivel1Usuario1**; Para los problemas del nivel 2 y el usuario 1 el nombre de la tabla será **PuntuacionesProblemasNivel2Usuario1**; Para los problemas del nivel 3 y el usuario 1 el nombre de la tabla será **PuntuacionesProblemasNivel3Usuario1**.

Para los problemas del nivel 1 y el usuario 2 el nombre de la tabla será **PuntuacionesProblemasNivel1Usuario2**; Para los problemas del nivel 2 y el usuario 2 el nombre de la tabla será **PuntuacionesProblemasNivel2Usuario2**; Para los problemas del nivel 3 y el usuario 2 el nombre de la tabla será **PuntuacionesProblemasNivel3Usuario2**.

Para los problemas del nivel 1 y el usuario 3 el nombre de la tabla será **PuntuacionesProblemasNivel1Usuario3**; Para los problemas del nivel 2 y el usuario 3 el nombre de la tabla será **PuntuacionesProblemasNivel2Usuario3**; Para los problemas del nivel 3 y el usuario 3 el nombre de la tabla será **PuntuacionesProblemasNivel3Usuario3**.

Para los problemas del nivel 1 y el usuario 4 el nombre de la tabla será **PuntuacionesProblemasNivel1Usuario4**; Para los problemas del nivel 2 y el usuario 4 el nombre de la tabla será **PuntuacionesProblemasNivel2Usuario4**; Para los

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

problemas del nivel 3 y el usuario 4 el nombre de la tabla será **PuntuacionesProblemasNivel3Usuario4;**

Al variar el contenido de estas tablas durante el uso de la aplicación estas tablas son creadas en el momento de abrir la aplicación por primera vez con una información inicial, e irán modificándose a medida que avanzan los usuarios a través de la aplicación.

Cada una de las tablas contendrá 10 filas, representando 10 problemas por nivel y para un usuario concreto. A continuación, un ejemplo de una de las tablas al ser inicializadas:

PuntuacionesProblemasNivel1Usuario1		
Idproblema	Puntuacion	Estado
1	0	1
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0

4-. Tabla NivelesUsuario[1|2|3|4]:

En esta tabla guardaremos los siguientes datos:

- Idnivel (Integer) : Código del nivel.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

- Estado (Integer) : Estado del nivel para el usuario.
 - Estado = 1. (Nivel actual accesible).
 - Estado = 2. (Nivel realizado no accesible).
 - Estado = 0. (Nivel no realizado no accesible).

En esta tabla se guarda la información necesaria; en este caso el estado de los niveles para cada uno de los usuario de la aplicación. Habrá una tabla por cada usuario para guardar dicha información con respecto al estado de los niveles.

Para el usuario 1 el nombre de la tabla será **NivelesUsuario1**.

Para el usuario 2 el nombre de la tabla será **NivelesUsuario2**.

Para el usuario 3 el nombre de la tabla será **NivelesUsuario3**.

Para el usuario 4 el nombre de la tabla será **NivelesUsuario4**.

Al variar el contenido de estas tablas durante el uso de la aplicación estas tablas son creadas en el momento de abrir la aplicación por primera vez con una información inicial, e irán modificándose a medida que avanzan los usuarios a través de la aplicación.

Cada una de las tablas contendrá 3 filas, representando los tres posibles niveles de los problemas de nuestra aplicación y para un usuario concreto. A continuación, un ejemplo de una de las tablas al ser inicializadas:

NivelesUsuario1	
Idnivel	Estado
1	1
2	0
3	0

5-. Tabla PuntuacionesTotales:

En esta tabla guardaremos los siguientes datos:

- Idusuario (Integer) : Código del usuario.

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

- Puntuacionmaxima (Integer) : Puntuación máxima del usuario.

En esta tabla se guarda la información necesaria; en este caso la puntuación máxima actual para cada uno de los usuario de la aplicación. Habrá una única tabla para guardar dicha información con respecto a las puntuaciones máximas actuales de cada usuario.

Al variar el contenido de esta tabla durante el uso de la aplicación esta tabla es creada en el momento de abrir la aplicación por primera vez con una información inicial, e irá modificándose a medida que avanzan los usuarios a través de la aplicación.

Esta tabla contendrá 4 filas, representando a cada uno de los 4 posibles usuarios de nuestra aplicación. A continuación, un ejemplo de la tabla al ser inicializada:

PuntuacionesTotales	
Idusuario	Puntuacionmaxima
1	0
2	0
3	0
4	0

Conclusiones

Elaboración de prototipos para el desarrollo de productos de tecnología social: ProMat5

El desarrollo del proyecto me ha permitido valorar la complejidad que implica el diseño y desarrollo de una aplicación completa en Android. Aunque a simple vista pueda parecer que la parte más complicada sería la implementación, la etapa más larga del proyecto ha sido el diseño y refinado de la mecánica de juego de la aplicación, que en nuestro caso se ha tratado de un proceso de prueba y error hasta encontrar una forma de juego cómoda y entretenida.

Durante el desarrollo se atraviesan las fases habituales de desarrollo: el proyecto empieza con ilusión y rápidamente se consiguen los primeros resultados, lo que produce satisfacción y motiva a seguir adelante. La frustración aparece con la corrección de errores y se acentúa mientras hay que buscar soluciones a los fallos de diseño, lo que se recompensa cuando la mecánica y el juego comienzan a ser jugables. Cuando lo único que queda es trabajo mecánico al igual que la mayoría de los proyectos se vuelve tedioso.

La creación de aplicaciones móviles está sujeta a la percepción de los usuarios. En nuestro caso, al ir nuestra aplicación dirigidas a niños; alumnos de 5º de primaria; aunque tengamos una aplicación robusta y estable, lo que interesa a nuestros usuarios es que tenga una interfaz sencilla y amable, a parte de intuitiva.

Este paso de acercamiento de la interfaz a los usuarios infantiles creo que ha sido la parte más tediosa de mi Trabajo Fin de Título, puesto que este tipo de usuario finales son muy exigentes y sinceros a la hora de cualquier tipo de valoración. Aún así, creo que he conseguido una interfaz bastante vistosa y que llama la atención, pese a no tener interfaces animadas o sonido.

Como ejemplo de ello, puedo decir que alguno de mis conocidos que han hecho uso de ella han estado entretenidos realizando ejercicios matemáticos y me han dado una buena crítica durante todo el desarrollo del producto y también sobre el producto final.

Este tipo de experiencia sobre el desarrollo de productos software orientados a la enseñanza de menores me ha resultado muy útil y gratificante como aprendizaje en el desarrollo de software bajo el entorno Android. En un principio pensaba que no sería capaz de realizar un trabajo completo y decente con respecto a este tipo de programación, pero poco a poco y tras muchas horas de trabajo, he conseguido adentrarme en el mundillo de la programación de los dispositivos móviles, y conseguir un producto final de Trabajo de Fin de Título acorde con las expectativas con las que lo veía desde un principio; a pesar de la falta de experiencia y conocimiento en esta materia nueva para mí.

Por último, me gustaría dar las gracias a los compañeros de la universidad que me han echado una mano. Y un agradecimiento muy especial a “MI FAMILIA”, ya que son ellos los que han tenido que aguantarme durante todo el proceso de desarrollo y me han permitido realizar este trabajo de manera eficaz con sus consejos, sus críticas de la evolución del proyecto, sus aportaciones sobre el mismo, etc... (¡GRACIAS!).

Bibliografía

<http://developer.android.com/index.html>

<https://class.coursera.org/android-001/lecture>

http://www.sgoliver.net/blog/?page_id=3011#

<http://www.sgoliver.net/blog/?p=1278>

<http://www.sgoliver.net/blog/?p=1611>

<http://www.sgoliver.net/blog/?p=1632>

<http://www.sgoliver.net/blog/?p=1646>

<http://windrealm.org/tutorials/android/android-listview.php>

<http://www.androidcurso.com/index.php/libros-android/el-gran-libro-de-android>

<http://www.androidcurso.com/index.php/tutoriales-android-fundamentos>

<http://www.androidcurso.com/index.php/tutoriales-android-avanzado>

<http://blog.vidasconcurrentes.com/android/creando-una-aplicacion-de-android-la-presentacion-2/>

<http://androidblog.reindustries.com/a-real-open-gl-es-2-0-2d-tutorial-part-1/>

<http://www.maestrosdelweb.com/editorial/curso-android-enviar-emails/>

<http://stackoverflow.com/questions/tagged/android>

EL GRAN LIBRO DE ANDROID, Autor: Tomás Gironés, Jesús Editorial: MARCOMO, S.A.