



*Asesorías y Tutorías para la Investigación Científica en la Educación Puig-Salabarría S.C.  
José María Pino Suárez 460-2 esq a Lerdo de Tejada, Toluca, Estado de México. 7223898475*

RFC: ATI120618V12

**Revista Dilemas Contemporáneos: Educación, Política y Valores.**

<http://www.dilemascontemporaneoseduccionpoliticayvalores.com/>

**Año: X      Número: 2.      Artículo no.:39      Período: 1ro de enero al 30 de abril del 2023.**

**TÍTULO:** El desarrollo de aplicaciones móviles para el reciclaje de códigos mediante la metodología METapp.

**AUTORES:**

1. Máster. Marco Antonio Checa Cabrera.
2. Máster. Rita Azucena Díaz Vásquez.
3. Máster. Andrés Roberto León Yacelga.

**RESUMEN:** El objetivo de esta investigación es presentar una propuesta para el desarrollo de una metodología denominada METapp, misma que está enfocada a trabajar con equipos pequeños y a la documentación elemental, lo cual otras aplicaciones no lo permiten. Para llevar a cabo este trabajo se utilizó encuestas y entrevista a expertos, además de un análisis bibliográfico profundo de las metodologías de aplicaciones existentes. Se concluyó que la metodología METapp es la aplicación más efectiva, ya que nos permite agilizar los procesos, reduciendo costos tanto en tiempo, recursos económicos y los medios que se necesitan para la construcción de aplicaciones móviles de calidad.

**PALABRAS CLAVES:** Metodología, recursos, aplicaciones, equipos.

**TITLE:** The development of mobile applications for code recycling using the METapp methodology.

**AUTHORS:**

1. Master. Marco Antonio Checa Cabrera.
2. Master. Rita Azucena Díaz Vásquez.
3. Master. Andrés Roberto León Yacelga.

**ABSTRACT:** The objective of this research is to present a proposal for the development of a methodology called METapp, which is focused on working with small teams and elementary documentation, which other applications do not allow. To carry out this work, surveys and interviews with experts were used, as well as an in-depth bibliographic analysis of existing application methodologies. It was concluded that the METapp methodology is the most effective application, since it allows us to streamline processes, reducing costs in terms of time, economic resources and the means needed to build quality mobile applications.

**KEY WORDS:** Methodology, resources, applications, equipments.

## INTRODUCCIÓN.

Las metodologías ágiles han ayudado de forma efectiva y rápida en el desarrollo de software con calidad, como lo manifiesta Martínez (2017), “todas las metodologías de desarrollo de software presentan modelos tendentes a producir software de la manera más eficiente y de la máxima calidad, atendiendo a ciertos criterios o factores considerados determinantes”, pero estas metodologías poseen ciertas características que aplicadas en el desarrollo de aplicaciones móviles permiten más bien limitar la planificación y estructuración del proyecto para hacerlo en tiempos y uso de recursos de acuerdo a lo que estas metodologías proponen (ver tabla 1), haciendo que su adecuación no ayude realmente en la creación de las APPs de calidad en el menor tiempo y costo (recursos y económicos) posible.

Tabla 1. Características que afectan el desarrollo de apps.

Característica	Metodologías Ágiles	Experiencia desarrolló APPs
Equipo de trabajo	(Monte, 2017), “Nueve personas”	Menos de 4, “un diseñador, uno o varios desarrolladores, backend”
Entregables	5 (SCRUM) pero (Roche, 2022) “conforme avanza el proyecto y comienzan a reflejarse los resultados, el product backlog cambia o aumenta en función a los siguientes objetivos”.	Tiempo de elaboración de entregables utilizarlo en programación.
Reciclaje de código	Genera Software Propietario, (Felix, 2011) “en el cual el autor no transmite ninguno de los derechos”	90% código fuente utilizado por los programadores es software libre. (Kang, 2018)

De acuerdo con lo mencionado anteriormente, se propuso desarrollar una metodología que cumpla con los principios propuesto “Individuos e interacciones sobre procesos y herramientas, software funcionando sobre documentación extensiva, colaboración con el cliente sobre negociación contractual, respuesta ante el cambio sobre seguir un plan” y principalmente, permita el trabajo en grupo, (León & Checa, 2022) “ya que demanda de una organización y planificación adecuadas para obtener los resultados deseados”.

Luego del análisis exhaustivo, se obtuvo como resultado la metodología METApp, que cumpliendo con su carácter de simplicidad, permite el desarrollo de aplicaciones móviles rápidas, por el uso de la técnica de reciclaje, entregables con formato simple, y por sobre todo, presentando fases reducidas que permitan el trabajo colaborativo de equipos de menos de uno a cuatro desarrolladores, con aplicación de pruebas durante todo el proceso de desarrollo.

Se trabajó con expertos dedicados en el desarrollo de apps, y se aplicó la metodología en el desarrollo de la aplicación radio online para evaluar la propuesta, involucrando a tres miembros de grupo evaluador para tener una apreciación más objetiva del método propuesto, y conocer los resultados, los más reales posibles dentro del contexto de la elaboración de aplicaciones.

Las metodologías de desarrollo de aplicaciones móviles existen, como por ejemplo, Mobile D, que son adaptaciones de las metodologías ágiles existentes como XP, SCRUM, entre otras, que según Reyes & Delgado (2021), “plantean diferentes reglas que se caracterizan por su grado de simplicidad y por su enfoque en la practicidad”, lo que permite un desarrollo exitoso de las iniciativas, pero que a igual con las metodologías con las cuales se basan, poseen las mismas características presentadas en la Tabla 1, y se considera principalmente, que el código aquí es original, se programa desde cero, para cumplir con el requisito de software propietario, pero que según Kang (2018) el 90% de código actualmente es abierto, por lo que esto constituye un cambio en la programación estándar.

La propuesta metodológica radica principalmente en el uso del reciclaje de código, técnica que todos los desarrolladores lo hacen de forma empírica, ya que a diferencia de la reutilización, que según Moreno (2019) es “utilizar clases y objetos de terceras personas”, y sobre las cuales se desconoce y tampoco interesa su implementación, pero sí su función es utilizarlo en las necesidades del proyecto; en cambio, el reciclaje va mucho más allá, se necesita adaptar el código (ya sea añadiendo, eliminando o modificando) a las necesidades del proyecto que el programador requiera; esto implica, no solo un objeto o clase, sino en ocasiones hasta de procesos completos acelerando considerablemente la tarea de la programación, y esto se logra por la naturaleza que tiene el software libre o código abierto del trabajo colaborativo de entre todos los miembros de la comunidad localizados en portales web que facilitan toda esta actividad, entre ellos: GitHub, PasteBin, HasteBin, Stack Overflow, en las cuales terceras personas presentan sus proyectos completos a la comunidad para que puedan ser utilizados como a bien pudiera por otros miembros, convirtiéndose en una ayuda significativa para los desarrolladores.

Las metodologías para apps existentes están enfocados a grupos de trabajo más amplios, pero a pesar de que Novoa-Hernández et al. (2019) “ofrecen un buen rendimiento e interacciones de trabajos en cortos períodos de tiempo para el desarrollo”, su documentación es excesiva (con dificultad en su redacción) dificultando de esta manera, que la creación de apps se haga lento, costoso (en recursos como en medios económicos) y con código sumamente complejo.

## **DESARROLLO.**

### **Materiales y métodos.**

Para el desarrollo de la metodología se aplicó la investigación descriptiva, misma que según WordPress (León et al., 2022) “permitió analizar e identificar las características del entorno con el fin de conocer la situación actual” de la creación de aplicaciones móviles por parte de los programadores.

**Análisis Previo.**

Se procedió en este paso a identificar las necesidades de los desarrolladores de APP en el uso de metodologías de desarrollo de software mediante encuestas a profesionales de la ciudad de Ibarra, Ecuador, que se encuentran elaborando APP y además de una investigación bibliográfica profunda, obteniendo así una posición teórica que determinó al final los elementos a considerar en la metodología METApp.

Una vez definido el diagnóstico anterior, se procedió a definir el conjunto de reglas, procesos y documentación que constituirá a METApp, y que permitirá un desarrollo ágil con equipos de trabajo reducido, y principalmente, considerando el reciclaje de código en la fase de elaboración de la APP.

**Evaluación.**

Finalmente, se evaluó la metodología resultante de acuerdo con el proceso propuesto de Méndez & Garrido (2006), que permita la identificación, cuantificación y valoración de los beneficios que permitió la nueva metodología en el desarrollo de una aplicación móvil; por ejemplo, con el objetivo de determinar si la aplicación de METApp es conveniente y beneficioso para los desarrolladores de APP.

El proceso de desarrollo de la metodología METApp inició con la entrevista dirigida a dos expertos en el tema: Ing. Rolando Cangás, Msc. (Gerente de la empresa SUMICOMPU) e Ing. Amparito Freire, Msc. (Directora del departamento de desarrollo de software de la U.E. Velasco Ibarra), ya que permite (Díaz et al., 2022) “obtener información crucial sobre las necesidades y problemas” específicos durante la creación de aplicaciones móviles; además, con la aplicación de una encuesta dirigida a 20 desarrolladores de software de la provincia de Imbabura, Ecuador, cuyos resultados se puede apreciar en la Figura 10.

## Resultados.

METApp se desarrolló basado en el manifiesto ágil, tomando como énfasis en la simplicidad, ya que se requiere una metodología lo más simple posible para desarrollar aplicaciones de calidad y usando como técnica principal el reciclaje de código en equipos de trabajo de uno a cuatro miembros y entregables con formato sencillo de cumplir, esto, como principales características de la propuesta.

La cultura sobre el cual está basado METApp es:

- **Transparencia.** Todos los involucrados conocen lo que se está haciendo.
- **Adaptación.** El equipo pueda adaptarse a los posibles cambios que se encuentran durante el desarrollo del proyecto.
- **Libertad.** Para que los desarrolladores afloren su creatividad con el uso de la técnica de reciclaje de código, y pueda usar, como a bien pudiera el código abierto, cumpliendo con los principios del movimiento de software libre.

Las tablas 2, 3 y 4 muestran los elementos que propone METApp:

Tabla 2. Roles.

ROLES	DESCRIPCIÓN
Cliente	Propone el proyecto y se encuentra relacionado en todas las fases de la metodología.
Líder	Encargado de la planificación y revisión de cada iteración y también parte integrante de un Team Developer.
Team Developer	Es el equipo desarrollador encargado de elaborar un Módulo Development.
Módulo Development	Es una de las partes de desarrollo de la aplicación propuesta, inicialmente dividida en la fase Preparatoria de METApp.

Tabla 3. Eventos.

EVENTOS	DESCRIPCIÓN
Initial Meeting	Reunión inicial entre todos los involucrados (Cliente, Líder, Team Developer), de aquí saldrá un bosquejo de la aplicación y la división del proyecto en varios Módulos Development.
Delivery Meeting	Reunión diaria del Team Development de cada Módulo Development, para conocer problemas, avances o nuevos requerimientos.
Release Meeting	Reunión final para gestionar la unión de los Módulos Development, posiblemente requiera un feedback a la fase de desarrollo.

Tabla 4. Entregables.

ENTREGABLES	DESCRIPCIÓN
Informe Initial Meeting	Se redacta al final de la reunión inicial (initial meeting), presentando los resultados obtenidos.
Informe Daily Report	Cada Team Developer elabora un informe diario con las observaciones, problemas y soluciones encontrados al desarrollar un Módulo Development.
Informe Delivery Meeting	Al finalizar su trabajo en su módulo asignado, cada Team Development presentará un informe final de módulo con los resultados obtenidos.
Informe Release Meeting	Todos los módulos son unidos para el lanzamiento final, previa reunión efectuada para el efecto, este es el informe que se entregará al final de este proceso.

La METApp consta de tres fases bien establecidas (ver figura 1): Preparatoria, Desarrollo y Release.



Figura 1. Fases de METApp.

La fase Preparatoria consiste en celebrar una reunión inicial con todos los elementos involucrados en el proyecto: el Cliente será el encargado de exponer su visión, requerimientos y necesidades a implementarlos en la APP; el Líder, en cambio, recomendará y compartirá sus recomendaciones en función de su experiencia. Una vez establecido un bosquejo, el Team Development empezará a realizar una estructura e interfaz previa en la herramienta preseleccionada con anterioridad, ya sea utilizando el wizard de la herramienta o utilizando un frontal localizado en plataformas como GitHub o PasteBin (reciclaje de código).

Es necesario indicar, que para establecer este bosquejo, los involucrados de la reunión deberán seguir las etapas establecidas como son (ver figura 2):

- **Análisis y Diseño.** Se analizará la visión y requerimientos propuestos por el Cliente y en forma conjunta con el líder se diseñará una estructura inicial.
- **Codificación y Pruebas.** El Team Developed acogerá la estructura inicial y le trasladará a la herramienta de desarrollo las técnicas indicadas anteriormente. Cabe indicar, que esta etapa tiene un feedback, mismo que será necesario en caso de añadir nuevos requerimientos o por si las pruebas realizadas lo exijan.
- **Evaluación.** Se evaluará el producto temporal obtenido por parte del Cliente, cumpliendo con sus expectativas iniciales.
- **Requerimientos.** La evaluación anterior por parte del Cliente puede generar nuevos requerimientos, por lo que se hace necesario volver al círculo de las etapas.

Finalmente, se procederá a definir los Módulos Development que trabajarán en la siguiente fase cada Team Development; se concluirá la reunión con la elaboración del entregable respectivo correspondiente a esta fase.



Figura 2. Fase Preparatoria.



En la fase Desarrollo (ver figura 3), en cambio, cada Módulo Development será desarrollado o programado por el Team Developed asignado para el efecto, siguiendo las etapas internas analizadas en la anterior fase, presentando el informe Daily Report al finalizar el día.

Otro punto importante es que cada Módulo Development estará en comunicación directa con otros, para que el desarrollo no se restrinja a un trabajo individual, sino más bien sea colaborativo, ya sea apoyándose entre ellos frente a problemas localizados, o en caso de ser necesario, compartir código o pruebas de unión temporal entre los diferentes módulos, o en su defecto, puede ser requerido el Cliente para hacer una observación del avance realizado una vez realizadas las pruebas que se realizan de una manera permanente en esta fase.

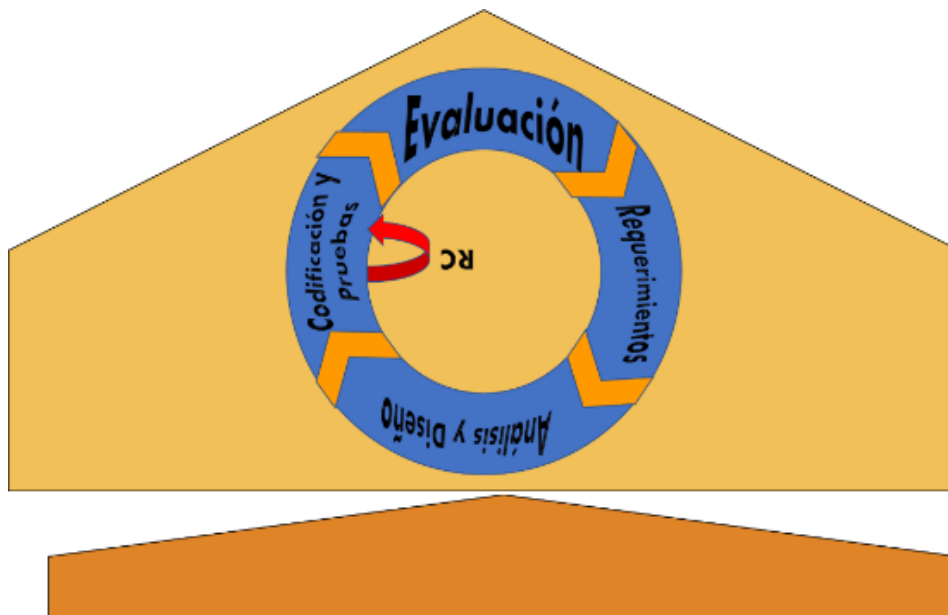
Al finalizar su trabajo, cada Team Developed presentará el informe Delivery Meeting en la que estarán presentándose los resultados obtenidos de la programación realizada en el módulo. La Fase concluye con el Delivery Meeting, que consiste en proceder a la unión de todas las partes con el direccionamiento del líder del proyecto.



**Figura 3.** Fase DESARROLLO.

La figura 4 muestra la fase final de Release, y aquí se celebra la reunión final Release Meeting, en la cual se entrega el producto final al Cliente, quien realizará su evaluación personal de satisfacción del

resultado de la metodología en el desarrollo de la APP; cabe recordar, que el Cliente estuvo en cada fase del desarrollo, observando los avances, por lo que muy difícilmente se requeriría un feedback hacia la fase de desarrollo en caso de existir nuevos requerimientos o problemas encontrados.



**Figura 4.** Fase Release.

Se realizará el cierre del proyecto con la entrega de los manuales y el archivo APK o AAB para su posterior publicación en el market, mismo que dependerá exclusivamente del Cliente.

Se puso a prueba la metodología METApp en condiciones reales de desarrollo de una APP por parte de un equipo desarrollador experto en la elaboración de la aplicación MW RADIO ONLINE solicitado por el Sr. Ronald Cupichamba Gerente General de Radio La Consentida en la ciudad de Ibarra, Ecuador.

Se aplicaron todas las fases en el desarrollo de la aplicación en forma detallada, contando con el apoyo del equipo de expertos. La tabla 5 muestra los equipos definidos después del Initial Meeting, así como la estructura (Figura 5), procesos (Figura 6) y bosquejo de interfaz inicial (Figura 7).

11  
 Tabla 5. Equipos.

No	Team Developer	Módulo Development
1	Ing. Amparito Freire	Interfaz principal, políticas, programación y publicidad
2	Ing. Rolando Cangás	Notificaciones
3	Ing. Rodrigo Varela	Reproducción de Video Streaming
4	Ing. Marco Checa	Reproducción de señal de Radio ONLINE

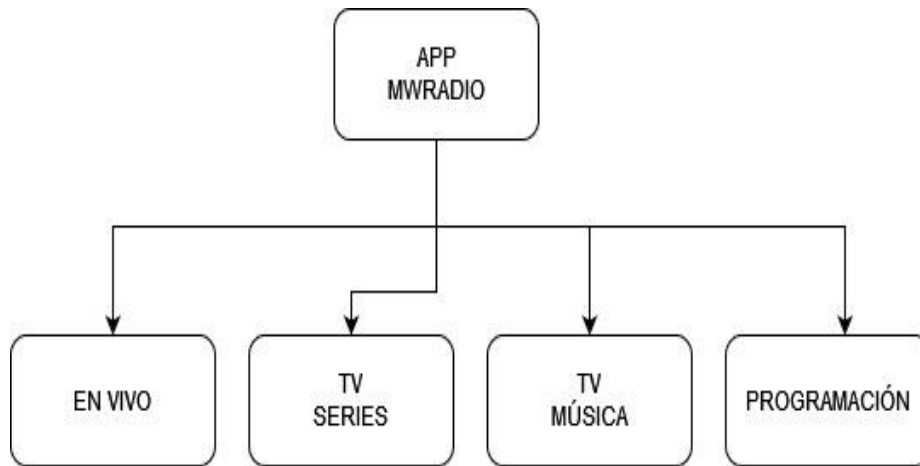


Figura 5. Estructura Inicial de la APP.

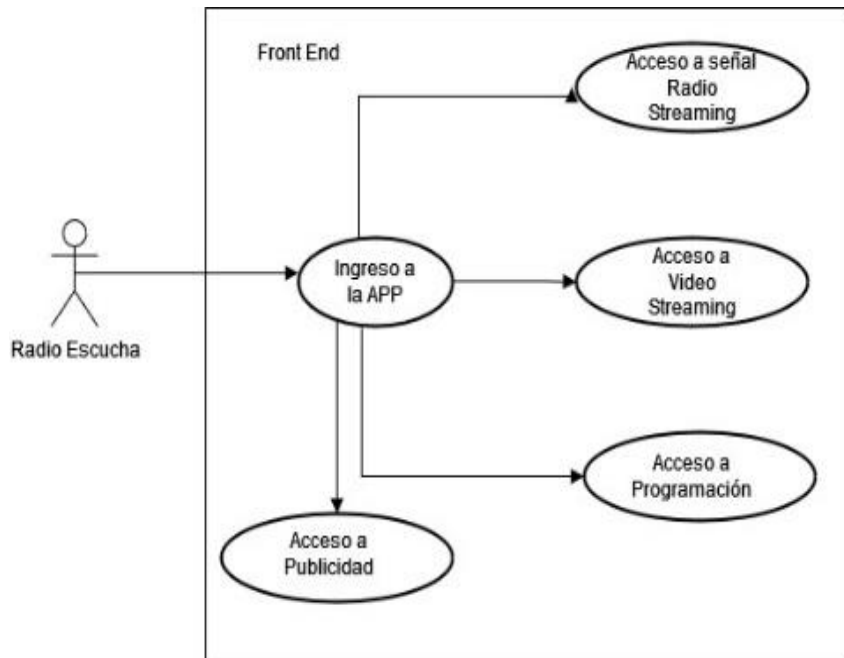


Figura 6. Procesos de la APP.

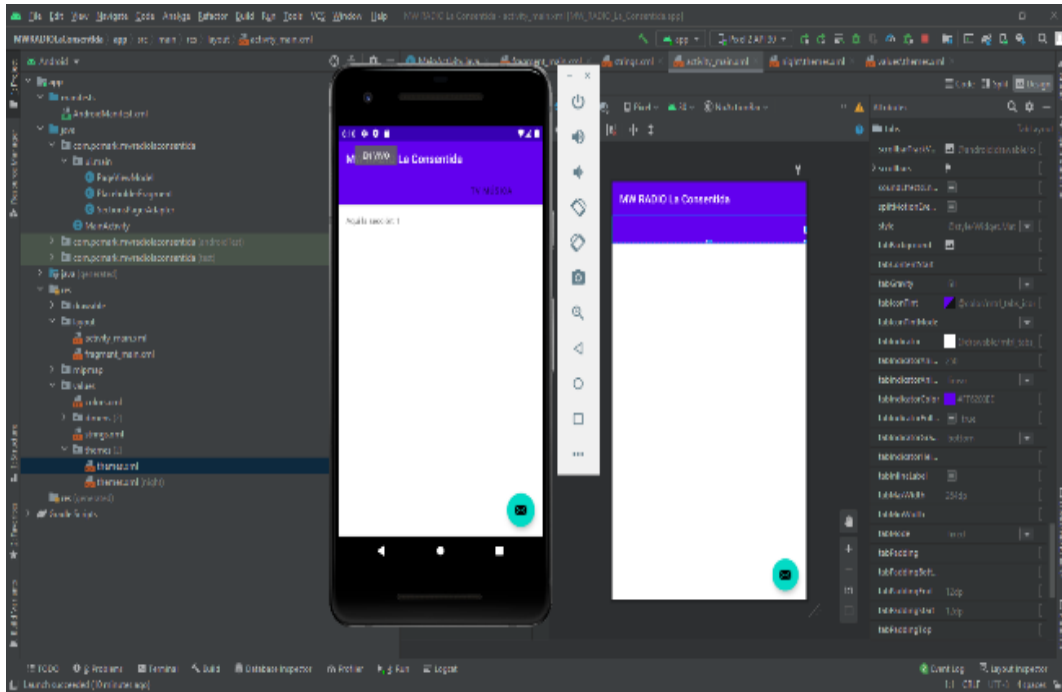


Figura 7. Interfaz Inicial obtenida tras el Initial Meeting.

## Resultados.

Una vez concluidas todas las fases, se obtuvo la aplicación en apenas 6 semanas y con 4 desarrolladores; la figura 8 presenta el producto final obtenido, y la figura 9 capturas de la publicación de la APP en el Play Store de Google.



Figura 8. Producto Final.

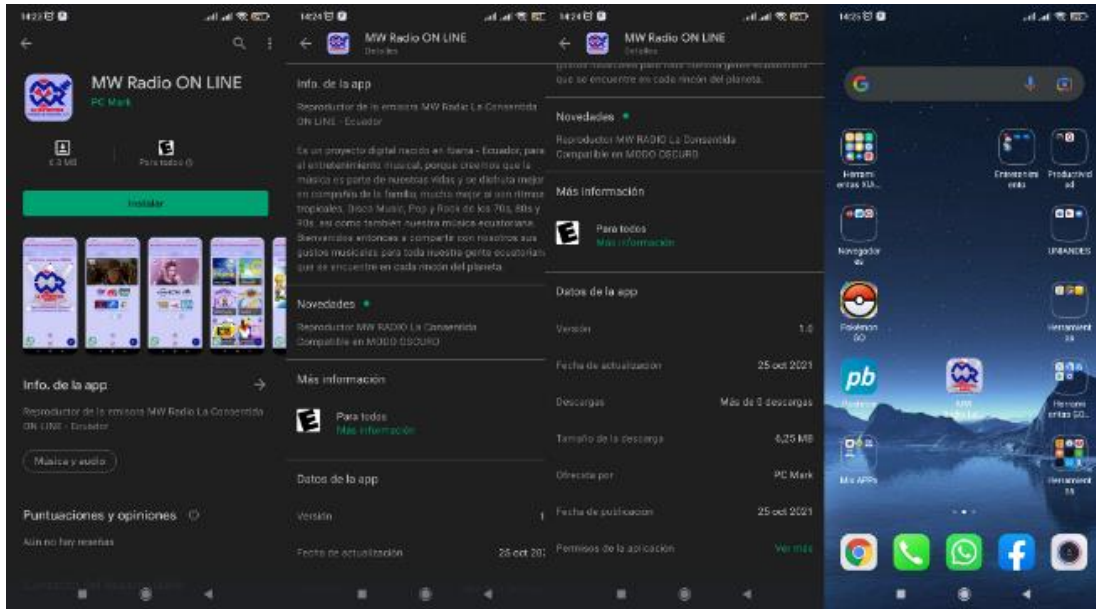


Figura 9. Capturas de la aplicación en el Play Store de Google.

Una vez concluido el desarrollo de la APP, se procedió a evaluar METApp a base de opiniones de expertos, aplicando encuesta basada en los criterios de Menéndez-Barzanallana (2018) expuestos en la tabla 6.

Tabla 6. Criterios de evaluación de METApp.

No	Criterio
1	Adecúa con los objetivos del proyecto.
2	Abarca el ciclo de desarrollo de software
3	Fases del ciclo de desarrollo integradas
4	Incluye proceso de validación
5	El ciclo de desarrollo cumple estrictamente los requerimientos del usuario
6	Existe una comunicación efectiva
7	Enfocado al usuario dinámicamente.
8	Relaciona responsables y resultados.
9	Extiende a proyectos de software variados.
10	Facilidad en aprendizaje.
11	Uso de la técnica de Reciclaje de Código.
12	Evoluciona con respecto al proyecto.
13	Actividades que mejora el proceso.

Estos criterios se evaluaron en un puntaje de 1 a 5, siendo uno No cumple Totalmente y cinco Cumple Totalmente. La figura 10 muestra los resultados de la evaluación.



Figura 10. Requisitos de una metodología.

Si bien se observa, que todos los criterios tienen índices realmente altos por parte de los expertos evaluados, demostrando de esta manera que METApp cumple con los requisitos deseables que debe tener una metodología, es necesario indicar, que de acuerdo a los tres puntos más altos, la propuesta tiene un enfoque hacia el cumplimiento de los objetivos (Criterio uno), identificando claramente los responsables de las actividades durante el desarrollo de la aplicación (criterio ocho), y tal vez, el más importante de los criterios es el hecho de que facilita el uso de la técnica de reciclaje de código (Criterio 11), como una herramienta de apoyo al programador tal vez más efectiva que las mismas herramientas CASE.

### **Discusión.**

Las metodologías ágiles existentes (XP y SCRUM), así como MOBILE D una metodología de desarrollo de aplicaciones móviles, están dirigidas al desarrollo de software con código original, propietario, y esto, los programadores en los momentos actuales no lo consideran adecuado para cumplir con el tiempo de desarrollo y con calidad, ya que la técnica del reciclaje de código ha ganado adeptos, porque se puede conseguir código fuente tanto de interfaces hasta código avanzado de procesos completos, que son presentados por terceros a la comunidad, cumpliendo con los objetivos

del software libre o código abierto.

Otro punto es la complejidad de formatos de entregables de las metodologías conocidas, en las cuales el programador dedica mucho tiempo en realizarlas, pudiendo aprovechar de mejor manera ese tiempo en la programación, algo que METApp ha minimizado en lo posible a formatos sencillos y fáciles de elaborar.

Finalmente, el número de integrantes del equipo de desarrollo en METApp se reduce considerablemente de 1 a 4 programadores a diferencia de las otras opciones que recomiendan más de 7.

## **CONCLUSIONES.**

METApp cumple con los objetivos propuestos de lograr una metodología que cumple con el manifiesto ágil, tomando como elemento principal la simplicidad, demostrado en sus fases y etapas que ofrece la propuesta, continuando con los eventos exclusivamente necesarios, así como de los entregables simples y sencillos de realizar, con equipos de trabajo reducidos y como columna vertebral la técnica de reciclaje de código.

Al reducir el tiempo de desarrollo, se reduce también los costos de desarrollo de las aplicaciones móviles, ya que los recursos necesarios no se utilizarán mucho tiempo, además que al utilizar software libre el costo por licencias se reduce a nulo en la práctica.

Los Módulos Development de METApp promueve un trabajo colaborativo en paralelo, en función a las diferentes partes que en el Initial Meeting fue dividido el proyecto, con libertad de interactuar entre ellos, incluso incluyendo al Cliente en caso de nuevos requerimientos que este solicite.

Al ser Android un sistema operativo que se actualiza de manera constante y en tiempos sumamente cortos, se hace necesario que la metodología también esté apta para volver a trabajar en una nueva versión en el menor tiempo posible, y METApp está preparada para este proceso con tan solo empezar desde la fase II, porque la primera fase ya se ha concluido, lo que se está haciendo es realizar el

lanzamiento de una nueva versión.

A futuro, se hace necesario una evaluación de METApp por una población de expertos más amplia, para tener un punto de vista más técnico y recibir observaciones que permitan la mejora de la metodología.

## REFERENCIAS BIBLIOGRÁFICAS.

1. Díaz, R., Acosta, J., & Checa, M. (2022). Vásquez, R. A. D., Espinoza, J. L. A., & Cabrera, M. A. C. (2022). Power bi como herramienta de apoyo a la toma de decisiones. Universidad y Sociedad, 14(S3), 195-207. <https://rus.ucf.edu.cu/index.php/rus/article/view/2949/2904>
2. Felix, F. (2011). Tipos de licencias de software: software libre, propietario y demás. (sitio web velneo). Obtenido de: <https://velneo.es/licencias-software-libre-propietario-otros/>
3. Kang, T. (2018). Open-source code recycling: Know your software supply chain. (sitio web techtarget). Obtenido de: <https://www.techtarget.com/iotagenda/blog/loT-Agenda/Open-source-code-recycling-Know-your-software-supply-chain>
4. León, A., & Checa, M. (2022). Uso de tableros Kanban como apoyo para el desarrollo de las metodologías ágiles. Universidad y Sociedad, 14(S2), 208-214. <https://rus.ucf.edu.cu/index.php/rus/article/view/2760/2735>
5. León, A., Acosta, J., & Checa, M. (2022). Wordpress para el diseño de sitios web en la fundación Santo Domingo de Guzmán. Universidad y Sociedad, 14(S3), 218-230. <https://rus.ucf.edu.cu/index.php/rus/article/view/2951/2906>
6. Martínez, A. (2017). Introducción a la Ingeniería del Software: Modelos de desarrollo de programas. Madrid: Delta Publicaciones.
7. Méndez, E., & Garrido, R. (2006). Modelo de evaluación de metodologías para el desarrollo de software. Caracas: Univ. Catol. Andres Bello. <http://biblioteca2.ucab.edu.ve/anexos/biblioteca/marc/texto/AAQ7365.pdf>



8. Menéndez-Barzanallana, R. (2018). Ingeniería del software. Metodologías habituales. Murcia, España: Departamento Informática y Sistemas. España: Universidad de Murcia.  
<https://www.um.es/docencia/barzana/IAGP/IAGP2-Metodologias-habituales.html>
9. Monte, J. (2017). Implantar scrum con éxito. Barcelona: Editorial UOC.
10. Moreno, P. (2019). Programación. Madrid: RA-MA.
11. Novoa-Hernández, P., Reyes, J., & Cedeño, J. (2019). Aplicación móvil inteligente para asistir el registro de actividades académicas en sistemas biométricos: una experiencia universitaria en el Ecuador. Revista Universidad y Sociedad, 11(2), 55-60.  
<http://scielo.sld.cu/pdf/rus/v11n2/2218-3620-rus-11-02-55.pdf>
12. Reyes, R., & Delgado, T. (2021). Modelo funcional de un laboratorio ciudadano de innovación digital. Revista Universidad y Sociedad, 13(1), 177-188. <http://scielo.sld.cu/pdf/rus/v13n1/2218-3620-rus-13-01-177.pdf>
13. Roche, J. (2022). Artefactos Scrum: las 3 herramientas clave de gestión. Elementos para la gestión de un proyecto de desarrollo de software incremento, Product Backlog y Sprint Backlog. (sitio web Deloitte). Obtenido de:  
<https://www2.deloitte.com/es/es/pages/technology/articles/artefactos-scrum.html>

## DATOS DE LOS AUTORES.

1. **Marco Antonio Checa Cabrera.** Máster Universitario en Ingeniería de Software y Sistemas informáticos. Docente de la Universidad Regional Autónoma de Los Andes Ibarra, Ecuador.  
Correo electrónico: [ui.marcocheca@uniandes.edu.ec](mailto:ui.marcocheca@uniandes.edu.ec)
2. **Rita Azucena Díaz Vásquez.** Máster Universitario en Ingeniería de Software y Sistemas Informáticos. Docente de la Universidad Regional Autónoma de Los Andes Ibarra, Ecuador.  
Correo electrónico: [ui.ritadiaz@uniandes.edu.ec](mailto:ui.ritadiaz@uniandes.edu.ec)

**3. Andrés Roberto León Yacelga.** Magíster en Gerencia Informática. Docente de la Universidad Regional Autónoma de Los Andes Ibarra, Ecuador. Correo electrónico:

[ui.andresleon@uniandes.edu.ec](mailto:ui.andresleon@uniandes.edu.ec)

**RECIBIDO:** 20 de octubre del 2022.

**APROBADO:** 17 de noviembre del 2022.