



Revista ingeniería e Innovación

ISSN: 2346-0466

ISSN: 2346-0474

revistaiei@correo.unicordoba.edu.co

Universidad de Córdoba

Colombia

Arregocés, Isabel; Ariza Coronado, Maicol; Camargo Martínez,  
Noryana; Díaz Hernández, Jesús; Gamarra, Margarita  
Integración de Scrum y RUP para el desarrollo de software  
de planes turísticos basado en preferencias de usuario  
Revista ingeniería e Innovación, vol. 10, núm. 1, 2022, Enero-Junio, pp. 21-32  
Universidad de Córdoba  
Colombia

- Número completo
- Más información del artículo
- Página de la revista en redalyc.org



# Integración de Scrum y RUP para el desarrollo de software de planes turísticos basado en preferencias de usuario

*Integration of Scrum and RUP for the development of tourist plan software based on user preferences*

Isabel Arregocés  Universidad de la Guajira,

Maicol Ariza Coronado  Universidad de la Costa

Noryana Camargo Martínez  Universidad de la Costa

Jesús Díaz Hernández  Universidad de la Costa

Margarita Gamarra  Universidad de la Costa

## RESUMEN

El desarrollo de aplicaciones móviles constituye una herramienta de fortalecimiento a diversos modelos de negocio, entre ellos, en el sector turístico. Sin embargo, notamos que las personas que viajan constantemente no cuentan con un mecanismo de recomendación para elegir los lugares a visitar y qué rutas podrían tomar dependiendo de sus preferencias y presupuesto. En este trabajo se propone una metodología híbrida para el desarrollo del software, el análisis de riesgo y una propuesta para el aseguramiento de la calidad. Además, se presenta un prototipo de la planificación del software que cuenta con unos patrones para generar diferentes planes turísticos y así los usuarios puedan tener una mejor experiencia y aprovechar su tiempo sin preocuparse antes de cada parada cual será la siguiente ruta. Esta aplicación contará con una página de inicio donde el usuario podrá registrar y crear su cuenta única donde realizará una encuesta que será utilizada como patrón para elegir los lugares, sitios, y rutas más cortos y económicos. El uso de la metodología híbrida permitió identificar los requerimientos funcionales y no funcionales, de manera que el software pudiera ajustarse a las necesidades de usuario bajo el cumplimiento de condiciones de tiempo y calidad.

**Palabras clave:** Sistemas de recomendación, metodología de desarrollo de software, scrum, planes turísticos.

## ABSTRACT

The development of mobile applications is a tool to strengthen different business models, including in the tourism sector. However, we noticed that frequent travelers do not have a recommendation mechanism to choose the places to visit and which routes they could take depending on their preferences and budget. In this work, a hybrid methodology for software development, risk analysis and a proposal for quality assurance is proposed. In addition, a software prototype is presented that has patterns to generate different tourist plans so that users can have a better experience and take advantage of their time without worrying what the next route will be before each stop. This application will have a home page where the user will be able to register and create their unique account where they will carry out a survey that will be used as a pattern to choose the shortest and most economical places, sites, and routes. The use of the hybrid methodology allowed identifying the functional and non-functional

Contaduría Pública, Universidad de la Guajira, Facultad de Ciencias Económicas y Administrativas, Riohacha, Colombia, [icarregoces@uniguajira.edu.co](mailto:icarregoces@uniguajira.edu.co), <https://orcid.org/0000-0001-5791-5533> \*jrfonseca@unicolmayor.edu.co,



Ingeniería de Sistemas, Universidad de la Costa, Dpto. Ciencias Computacionales y Electrónica, Barranquilla, Colombia, [mariza30@cuc.edu.co](mailto:mariza30@cuc.edu.co), <https://orcid.org/0000-0001-5001-8012> \*



Ingeniería de Sistemas, Universidad de la Costa, Dpto. Ciencias Computacionales y Electrónica, Barranquilla, Colombia, [ncamargo5@cuc.edu.co](mailto:ncamargo5@cuc.edu.co), <https://orcid.org/0000-0002-0386-9583> \*



Ingeniería de Sistemas, Universidad de la Costa, Dpto. Ciencias Computacionales y Electrónica, Barranquilla, Colombia, [jdiaz@cuc.edu.co](mailto:jdiaz@cuc.edu.co), <https://orcid.org/0000-0002-7564-493> \*



**Fecha recepción:**  
**Fecha aceptación:**



requirements, so that the software could be adjusted to the user's needs under the fulfillment of time and quality conditions.

*Commons Attribution License, que permite el uso ilimitado, distribución y reproducción en cualquier medio, siempre que el autor original y la fuente se acreditan.*

**Keywords:** Recommendation systems, software development methodology, scrum, tourism plans.

## INTRODUCCIÓN

El uso creciente de dispositivos móviles ha generado que éste se convierta en una herramienta de consulta permanente por parte de los usuarios. El desarrollo de diversas aplicaciones móviles permite que muchos modelos de negocios incorporen sus servicios en este tipo de software para que el usuario pueda acceder a ellos desde donde se encuentre. Uno de los sectores que se ha visto beneficiado con la implementación de nuevas tecnologías es el sector turístico (Sánchez Jiménez & Fernández Allés, 2018), con el desarrollo de aplicaciones que van desde trazar una ruta, búsqueda de alojamiento, información sobre sitios turísticos hasta reconstrucción de sitios históricos con ayuda de realidad virtual.

Sin embargo, la exposición a toda esta gran cantidad de información genera una gran dificultad para el usuario y un gasto de tiempo en obtener una solución que se ajuste a sus necesidades. Por esta razón se hace necesario crear una aplicación móvil basada en las preferencias y restricciones del usuario, que recomiende la mejor ruta y destino en un tiempo óptimo.

Existen trabajos recientes que presentan diversas opciones de incorporación de tecnologías de la información en el sector turismo (Solano Barliza, 2021). El trabajo

desarrollado por (Merino-Sánchez et al., 2020) muestra la planificación del desarrollo de una aplicación móvil para fomentar y dilatar local e internacionalmente el turismo en la parroquia Bayushig perteneciente al cantón Penipe de la Provincia de Chimborazo en Ecuador. Para esto utilizaron la metodología de planificación conocida como Métrica V3.

Otra aplicación móvil en el sector turístico fue desarrollada por (Chala Plata et al., 2020), la cual facilita a los usuarios hacer turismo en la costa caribe colombiana. Esta aplicación tiene una arquitectura basada en la nube y está basada en el contexto y localización de los usuarios para generar resultados de la búsqueda.

En el trabajo desarrollado por (Quezada-Sarmiento et al., 2018) aplica metodologías ágiles para el desarrollo de aplicaciones web para la promoción de rutas ecoturísticas en la Amazonía sur de Ecuador. También incorporan el uso de redes sociales y la identificación de dos nuevas rutas ecoturísticas que diversifican la oferta turística de la zona.

Los autores del trabajo publicado en (Norman Anuar & Kamal Othman, 2020) integran el diseño centrado en el usuario y las metodologías ágiles para el diseño de un sistema de información de turismo cultural.

Se solicitó a cinco usuarios que expresaran sus ideas sobre cómo preferían la aplicación y las funciones que tendría, como centro de información sobre el patrimonio cultural en Sarawak mientras fomenta la conservación y preservación de la cultura material. El estudio encontró que los usuarios prefieren un diseño simple y agradable con énfasis en la representación visual y navegación intuitiva.

Existen muchos factores que pueden hacer de esta aplicación móvil un éxito, si se tienen en cuenta los lugares turísticos y cada una de sus características como precios, calificación y ubicación. Este trabajo tiene como objetivo presentar una metodología de desarrollo de software aplicada a la construcción de un prototipo de aplicación móvil para la generación de planes turísticos basados en las preferencias de usuario, dando a conocer información valiosa, precisa y única de cada uno de los lugares turísticos.

1. METODOLOGÍA

Desarrollar un buen software depende de múltiples actividades y etapas, donde el impacto de elegir la mejor metodología para un equipo, en un determinado proyecto es trascendental para el éxito del producto y la optimización de los recursos. A continuación, se exponen las dos metodologías seleccionadas para ser integradas en el proceso de desarrollo del software de recomendación turística.

1.1. Metodologías ágiles

Las metodologías ágiles ofrecen una solución adecuada para una gran cantidad de proyectos que deben realizarse con limitaciones de tiempo. Una de las cualidades principales de las metodologías ágiles es su sencillez, tanto en su aprendizaje como en su aplicación, lo que facilita su implantación en un equipo de desarrollo. En la Tabla 1 se exponen las ventajas y desventajas de las metodologías ágiles.

Tabla 1. Ventajas y desventajas de las metodologías ágiles

| Ventajas                                                                                                         | Desventajas                                   |
|------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|
| Rápida respuesta a cambios de requisitos a lo largo del desarrollo del proyecto gracias a su proceso iterativo.  | Se tiene poca documentación del diseño.       |
| El cliente puede colaborar, puede observar cómo va avanzando el proyecto y dar sus opiniones sobre su evolución. | Se presenta gran dependencia de las personas. |
| Buscan la simpleza a la hora de trabajar para así eliminar el trabajo innecesario y ahorrar tiempo.              | Difícil la re-utilización del código.         |

Fuente: elaboración propia

La producción de software que necesitamos debe ser un desarrollo ágil y entrega rápida,

sin detenerse demasiado en la documentación. Sin embargo, el no contar con la documentación necesaria requiere inversiones mayores no presupuestadas en tiempo y recursos. Por lo tanto, una metodología completamente ágil no sería adecuada en este caso. Debido a esto se propone integrar una metodología tradicional y una ágil para adaptarlas en un nuevo proceso de desarrollo donde se completen dichas metodologías con enfoques diferentes, pero complementarios. Esta metodología híbrida está siendo utilizada de manera eficiente y exitosa en otros contextos (Mariani et al., 2010) (Schuh et al., 2018).

Para el desarrollo del prototipo de software de planeación turística se ha seleccionado la metodología ágil Scrum y la metodología de

base tradicional RUP (Rational Unified Process) para generar un modelo híbrido que permita satisfacer los requerimientos del software bajo restricciones adecuadas de tiempo, costo y calidad. A continuación, se mencionan las principales características de estas dos metodologías.

## 1.2. Metodologías ágiles Scrum y RUP

En la Tabla 2 se muestra una comparación entre las metodologías de desarrollo de software ágiles conocidas como Scrum y RUP (Sommerville, 2015) (Pressman, Roger S., 2015)

Tabla 2. Comparación entre Scrum y RUP

|                  | SCRUM                                                                                                                                  | RUP                                                                                                                                    |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| Enfoque          | Iterativo                                                                                                                              | Iterativo                                                                                                                              |
| Ciclos           | Cada sprint es un ciclo completo                                                                                                       | Tiene un solo ciclo con cuatro fases                                                                                                   |
| Fases            | La única fase es el sprint, con duración de 2 a cuatro semanas. Existe una fase previa de planeación donde se define el sprint backlog | Incepción o iniciación<br>Elaboración<br>Construcción<br>Transición                                                                    |
| Planeación       | No hay un plan para el proyecto completo. Cada plan de la siguiente iteración se determina al final de la iteración actual.            | Se utiliza un plan de proyecto formal, asociado con múltiples iteraciones. El plan incluye una fecha final, y tiene hitos intermedios. |
| Alcance          | Utiliza el product Backlog, el cual es evaluado al final de cada iteración (Sprint)                                                    | El alcance es definido antes del inicio del proyecto. El alcance puede revisarse durante el proyecto.                                  |
| Tipo de proyecto | Recomendado para proyectos rápidos y organizaciones que no dependen de una fecha límite.                                               | Recomendado para proyectos a nivel empresarial, a largo plazo y de complejidad media a alta.                                           |

Fuente: elaboración propia

Después de comparar estas dos metodologías y analizar lo que ofrece cada metodología, fueron adaptadas a los requerimientos del proyecto, siguiendo la propuesta desarrollada por (Mariani et al., 2010). Scrum se utilizará para gestionar los procesos del proyecto. Esta metodología resulta confiable para definir un marco de trabajo para la gestión y el desarrollo de software basándose en un proceso iterativo e incremental. Por otra parte, los cambios de requerimientos son bastante frecuentes en este contexto debido a los diversos intereses de los usuarios y los procesos de innovación implícitos en el contexto turístico. De esta forma, si se adopta esta metodología ágil, adaptarse a los cambios es más accesible.

Por otra parte, la metodología RUP se utilizará para tomar los principios para definir los artefactos ya que esta propone realizar unos encuentros donde el usuario y el equipo de análisis se reúnen y se presentan los requisitos deseados y sus restricciones. Este análisis inicial conduce a la organización de las etapas del desarrollo del software, con una participación del cliente. La metodología RUP propone una estructura estática y una dinámica para caracterizar el software.

### 1.3. Estructura estática

Define la organización lógica de los elementos del proceso, incluyendo actividades, disciplinas, artefactos y roles. Un proceso describe quién, qué, cómo y cuándo. A través de los roles, se expresa quién (una persona o un grupo) está

realizando el trabajo. Una actividad describe cómo lo está llevando a cabo el trabajo y generalmente se trata de la creación o actualización de un artefacto. Un artefacto es el resultado de ese trabajo y es un segmento de información que es producida, modificada o utilizada dentro del proceso. Uno de los principales artefactos que propone la metodología RUP son los diagramas y la arquitectura. Algunos artefactos son tomados como entradas a un proceso para ejecutar una actividad y producir un resultado o un artefacto de salida. Para definir el cuándo del proceso, se precisa entonces un workflow o flujo de trabajo. El workflow es la secuencia de actividades y la interacción entre los diferentes roles, ya que los roles, actividades y artefactos por sí solos no constituyen un proceso.

### 1.4. Estructura dinámica

Esta estructura divide el proyecto en cuatro fases: Inicio, Elaboración, Construcción y Transición. Cada una de estas fases tiene unos objetivos definidos, como se muestra a continuación.

#### 1.4.1. Fase de inepción o iniciación:

1- Modelo del proceso de negocio.

En esta fase el equipo de trabajo conoce el funcionamiento de la empresa y sus procesos. Los objetivos de esta etapa son:

- Entender la estructura y el funcionamiento de la organización donde se implementará el sistema.
- Entender el problema y/o necesidades actuales e identificar potenciales mejoras.

- Lograr que los clientes, usuarios finales y desarrolladores tengan un entendimiento del contexto.

## 2- Requerimientos (Product Backlog).

En esta línea se fijan los requerimientos, de modo que el cliente y los usuarios finales deben comprender y aceptar los requerimientos especificados. Los objetivos de esta etapa son:

- Establecer y mantener un acuerdo entre clientes y otros involucrados sobre el sistema.
- Proveer a los desarrolladores un mejor entendimiento de los requerimientos.
- Definir el alcance y limitaciones del sistema.
- Proveer una base para estimar costos y tiempo de desarrollo.
- Definir una interfaz de usuario, enfocadas a las necesidades y metas del usuario.

### 1.4.2. Fase de elaboración:

## 3- Análisis y diseño.

En esta actividad se especifican y analizan los requerimientos y se describen sobre cómo se van a implementar en el sistema. Los requerimientos son plasmados en distintos artefactos generados por los analistas, por ejemplo, a través de casos de uso. Estos artefactos son presentados al usuario posteriormente y este puede realizar alguna modificación que es ajustada por los analistas. Esto permite una buena retroalimentación al realizar la solicitud y la propuesta de los requerimientos.

Luego de la aceptación del usuario, se pasa a la siguiente etapa donde se documentan los requerimientos obtenidos y ajustados en los artefactos.

Artefactos por cada uno de los Sprint o

módulos:

- Iteración o Sprint.
- Sprint Backlog o requerimientos del sprint
- Diagrama de componentes.
- Plan de pruebas
- Diagrama de despliegue o implementación.

A través de esta estructura de trabajo, se observa la integración de ambas metodologías. Se mantuvo la base del Scrum, y se consideró cada Sprint como un proyecto independiente, de manera que cada Sprint se convierte en único ciclo donde se pudo agregar el proceso RUP para el área de análisis, donde presenta una gran fortaleza. En la fase de análisis hay un consenso entre el cliente y el equipo de trabajo, con la finalidad de generar la documentación e integrarla en los artefactos correspondientes. Los artefactos son utilizados por el equipo desarrollador, quienes se encargan de realizar la planificación y se organizan en equipos, bajo la línea del modelo Scrum.

## 2. SOFTWARE PROPUESTO

La propuesta de este trabajo es diseñar un software de una App móvil del cual se pueda descargar mediante un código QR y colocarlos en los sitios que los turistas o cualquier persona visitan con mayor frecuencia. Este software se podrá ejecutar creando algunos puntos con personas que promocionen el turismo y tengan una foto del QR y les generen convencimiento a las personas para motivarse más a conocer la ciudad. En esta aplicación se le pedirán ciertos datos al usuario para tener una base

de datos de todas las personas y poder contactarlas ante alguna nueva sugerencia acerca los lugares que puede visitar.

Al ingresar los datos, se realiza una breve encuesta que será con el que el algoritmo podrá recomendar que lugares visitar, ya sea teniendo en cuenta el gusto de los lugares de las personas como museos, paisajes, urbanización y demás o el tiempo que tengan estas personas en la ciudad para conocer la mayor cantidad de lugares, otro aspecto importante también sería el presupuesto ya que permitirá saber a las personas aproximadamente el total de sus posibles gastos. También para promover el orden, se colocará un QR en cada una de las entradas de los sitios turísticos para la comodidad y una fácil movilización de las personas durante los recorridos. A continuación, se mencionan las características principales para el desarrollo del software propuesto:

- Enfoque a la creación de prototipos El Modelo de prototipos, hace parte de los modelos de desarrollo evolutivo. El prototipo se caracteriza por construirse en poco tiempo y no se debe utilizar muchos recursos. Su finalidad es presentar un diseño preliminar de la versión final, para que el cliente pueda hacer modificaciones.

- Lenguajes de programación PHP (Hostinet, 2022): Es un lenguaje de programación de uso general que se adapta especialmente al desarrollo web. Fue creado inicialmente por el programador danés-canadiense Rasmus Lerdorf en 1994. En la actualidad, la implementación de referencia de PHP es producida por The PHP Group.

JavaScript (MDN, 2022): es un lenguaje de programación ligero, interpretado, o

compilado justo-a-tiempo con funciones de primera clase. Si bien es más conocido como un lenguaje de scripting para páginas web, y es usado en muchos entornos fuera del navegador, tal como Node.js, Apache CouchDB y Adobe Acrobat. JavaScript es un lenguaje de programación basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa.

Java (Java, 2021): Es un lenguaje de programación y una plataforma informática que fue comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán, probablemente, a menos que tengan Java instalado y cada día se crean más. Java es rápido, seguro y fiable.

## **2.1. Identificación del equipo de trabajo**

Para la ejecución de este proyecto es necesario integrar personal especializado tanto en la parte de programación como en el área de turismo, ya que la aplicación debe cumplir no solo con requisitos funcionales, sino que debe ser atractiva para el turista, quien es el usuario final.

Jefe de proyecto: este rol se encarga de planificar, ejecutar y monitorear el proceso de desarrollo de software. Se encarga de liderar la toma de decisiones que van orientadas al cumplimiento de los objetivos de cada fase y del proceso como tal.

Desarrollador web: es un programador encargado de desarrollar aplicaciones de la World Wide Web o aplicaciones distribuidas en red que se ejecutan mediante HTTP desde un servidor web a un

navegador web.

**Experto en base de datos:** Es el responsable por la Administración de las Bases de Datos. Administra las tecnologías de la información y la comunicación, siendo responsable de los aspectos técnicos, tecnológicos, científicos, inteligencia de negocios y legales de bases de datos.

**Comunicador social:** puede trabajar para medios, empresas y organizaciones sociales. En este sentido, puede laborar creando y produciendo mensajes e informaciones para el desarrollo del software turístico.

**Gestor cultural:** persona con conocimiento especializado en arte, historia, cultura e idiomas, que se dedican a promover, incentivar, diseñar y realizar proyectos donde se aprecie y se valore los lugares emblemáticos y culturales de las ciudades.

**Turismólogo:** profesional que conoce, investiga, estudia y analiza el turismo en su totalidad, considerando su campo de análisis como una ciencia social en constante desarrollo metodológico y relacionado con las ciencias económicas.

Como estructura del equipo de trabajo se utilizará la estructura piramidal o jerárquica, ya que debe primar el orden y tener un líder que integre las diferentes disciplinas. Si se desea hacer una gestión de configuración al proyecto, la aprobación o la orden debe de venir únicamente de arriba, así se respetarán los roles y las responsabilidades asignadas.

## 2.2. Estimación de costos

El presupuesto planificado de un proyecto debe ser lo más preciso posible para el éxito del proyecto y su finalidad. La estimación de costes es una de las formas más eficaces del gestor de proyectos para planificar un presupuesto preciso.

El presupuesto de un proyecto afecta de manera directa a casi todas las facetas de este, por lo que es una de las responsabilidades más importantes del gestor de proyectos. Un presupuesto mal diseñado conduce a una asignación inadecuada de los activos, a expectativas poco realistas y a un posible fracaso del proyecto.

## 2.3. Análisis de riesgos

Teniendo claro cada uno de los detalles y sucesos posibles del proyecto, cabe resaltar la posible aparición de los errores. A pesar de que el proyecto esté muy bien elaborado, la aparición de riesgos no deja de ser posible aunque tenga un porcentaje bajo de probabilidad, puede pasar ya sea por una mala interpretación del uso y esto causar desinterés de la App o también que la falta de conocimiento de la tecnología y pensamiento lógico no sea suficiente, riesgo de una caída del sistema, olvido de datos, memoria insuficiente y por ende una falla en la ejecución de App móviles, y otros posibles casos que se puedan presentar durante la ejecución continua del proyecto. En la Tabla 3 se registran los principales riesgos del proyecto su nivel de peligro y las posibles consecuencias si llegara a ocurrir. Estos riesgos están relacionados con posibles fallas en el soporte del software.

Tabla 3. Peligros, riesgos y consecuencias en el desarrollo del software

| Peligro        | Riesgo                              | Consecuencia                                                                  |
|----------------|-------------------------------------|-------------------------------------------------------------------------------|
| Insignificante | Olvido de datos                     | El dejar de utilizar la App o tener que realizar un proceso de recuperación.  |
| Moderado       | Memoria insuficiente                | Error de ejecución de la App o que no sea posible su descarga.                |
| Dañino         | Fallas del sistema                  | No poder utilizar la App y la posible pérdida de acogida de los usuarios.     |
| Dañino         | Falta de conocimiento en tecnología | No poder manipular la App de forma correcta y no poder explotar su potencial. |

En general, los resultados de este metaanálisis mostraron un notable avance en la comprensión de los mecanismos utilizados por *S. maltophilia* para la degradación y/o transformación de hidrocarburos y metales pesados, pues los estudios seleccionados en el primer periodo de tiempo en su mayoría concluyeron en la necesidad de realizar estudios a profundidad en el futuro con tecnologías más avanzadas a su época para comprender de mejor manera los procesos evidenciados en ese entonces. Por el contrario, los estudios seleccionados en el segundo periodo de tiempo, especialmente posteriores al año 2010 mostraron una descripción más detallada de los mecanismos evidenciados en los resultados de sus experimentos.

**2.4. Sistema de control de versiones para la gestión de cambios del proyecto.**

La gestión de cambios es una parte importante dentro del proceso de desarrollo de software, especialmente en aquellos sectores donde la innovación y el cambio son frecuentes, como en el contexto del turismo. Tener un adecuado control de versiones, o "control de código fuente", es

fundamental para la gestión de cambios. El control de versiones permite rastrear y gestionar los cambios en el código de software y facilita el trabajo en equipo. Los sistemas de control de versiones son herramientas de software que ayudan a los equipos desarrolladores a gestionar los cambios en el código fuente a lo largo del tiempo de forma más rápida e inteligente (Bitbucket, 2022).

La mejor alternativa en este caso sería utilizar GitHub ya que podemos obtener varias ventajas como:

- Disponer de un completo historial de cambios a largo plazo de todos los archivos. Esto quiere decir todos los cambios realizados por muchas personas a lo largo de los años. Tener el historial completo permite volver a las versiones anteriores para ayudar a analizar la causa de los errores.
- Creación de ramas y fusiones. Permite tener múltiples flujos de trabajo independientes los unos de los otros al tiempo que ofrece la facilidad de volver a fusionar ese trabajo.
- Trazabilidad. Tener el historial del código al alcance puede permite hacer cambios

adecuados que estén acordes con el diseño previsto a largo plazo del sistema. Esto es importante para trabajar de manera eficaz con código heredado y es esencial para que los desarrolladores puedan calcular el trabajo futuro con precisión.

## 2.5. Aseguramiento de la calidad

Para garantizarle a los usuarios finales, en nuestro caso los turistas, un producto desarrollado bajo estándares de calidad y ajustados totalmente a sus necesidades se han planteado un conjunto de actividades que aseguran que el proceso de desarrollo de un software sea evaluado bajo criterios de calidad, con una correcta implementación de procesos, requerimientos y estándares de desarrollo.

Para ello, se ha propuesto la continua actualización de la información, como, por ejemplo; actualizar lugares nuevos, lugares disponibles, nuevas actividades para los turistas, etc. Además, otras actividades a ejecutar con fines de mantener la calidad son:

- Revisar los productos para verificar los procedimientos aplicados.
- Descubrir errores en la funcionalidad y lógica del software.

## 3. CONCLUSIONES

Una correcta definición de los requisitos generales del sistema desde el inicio del proyecto facilita la implantación de la creación de prototipos como alternativa de diseño. Conocer el contexto cultural de la región y las necesidades del usuario final son fundamentales para la generación de

estrategias de diseño del software de planeación turístico. Adicionalmente, la integración de las metodologías Scrum y RUP permite incorporar las ventajas de ambas metodologías, mejorar la organización de los equipos, tener una correcta planeación por fases y definir actividades de acuerdo con los roles asignados.

Finalmente, el desarrollo del proyecto guiado por las metodologías Scrum y RUP, la asignación de roles adecuados, la utilización de herramientas para la estimación de costos y control de versiones y un correcto análisis de riesgos, resultan ser los elementos adecuados para la implementación del software de planificación turística.

## REFERENCIAS

- [1]. Bitbucket. (2022). Qué es el control de versiones | Atlassian Git Tutorial. Atlassian Bitbucket. <https://www.atlassian.com/es/git/tutorials/what-is-version-control>
- [2]. Chala Plata, M. E., Cerpa Jiménez, J. J., & González Pérez, B. J. (2020). Diseño e implementación de una aplicación móvil para impulsar el turismo en el caribe colombiano [Barranquilla, Universidad del Norte, 2020]. <http://hdl.handle.net/10584/9276>
- [3]. Hostinet. (2022). PHP 8.0 Disponible en Hostinet ¿Ya lo has Probado? <https://www.hostinet.com/formacion/hosting-alojamiento/php80-disponible/>
- [4]. Java. (2021). ¿Qué es Java y para qué es necesario?

- [https://www.java.com/es/download/help/whatis\\_java.html](https://www.java.com/es/download/help/whatis_java.html)
- [5]. Mariani, A. :, Florencia - Okabe, M., & Director, E. (2010). PDSM: Proceso de Desarrollo de Software Mixto combinando RUP y SCRUM. <http://sedici.unlp.edu.ar/handle/10915/4034>
- [6]. MDN. (2022). JavaScript | MDN. <https://developer.mozilla.org/es/docs/Web/JavaScript>
- [7]. Merino-Sánchez, C. G., Játiva-Brito, M. E., Buenaño-Pesantez, C. V., & Coronel-Maji, F. M. (2020). Planificación del desarrollo de una aplicación móvil para fomentar el turismo en Bayushig – Ecuador - Dialnet. Polo Del Conocimiento: Revista Científico - Profesional, 5(8), 130–150. <https://dialnet.unirioja.es/servlet/articulo?codigo=7554382>
- [8]. Norman Anuar, N., & Kamal Othman, M. (2020). Integrating User-Centered Design with the Agile Software Development Methodology for a Cultural Heritage Information System. Interaction Design and Architecture(s) Journal, 45(1), 207–225.
- [9]. Pressman, Roger S., B. R. M. (2015). Software Engineering A Practitioner's Approach. McGraw-Hill Education, 2 Penn Plaza, New York, NY 10121, 1(4), 53. [www.mhhe.com/pressman](http://www.mhhe.com/pressman).
- [10]. Quezada-Sarmiento, P. A., Suasnavas-Rodriguez, M. G., Chango-Canaveral, P. M., Gonzaga-Vallejo, C., Enciso, L., & Calderón-Cordova, C. A. (2018). Used of social networks and web application to design and promote the ecotourism route in the Southern Amazon of Ecuador. 2018 13th Iberian Conference on Information Systems and Technologies (CISTI). <https://ieeexplore-ieee-org.ezproxy.cuc.edu.co/document/8399399/>
- [11]. Sánchez Jiménez, M., & Fernández Allés, M. T. (2018). Revisión teórica de la relevancia de las nuevas tecnologías de la comunicación (TIC) en el sector turístico - Dialnet. TURRYDES: Revista Sobre Turismo y Desarrollo Local Sostenible, 11(24), 1–21. <https://dialnet.unirioja.es/servlet/articulo?codigo=7845416>
- [12]. Schuh, G., Rebentisch, E., Riesener, M., Diels, F., Dölle, C., & Eich, S. (2018). Agile-waterfall hybrid product development in the manufacturing industry - Introducing guidelines for implementation of parallel use of the two models. IEEE International Conference on Industrial Engineering and Engineering Management, 2017-December, 725–729. <https://doi.org/10.1109/IEEM.2017.8289986>
- [13]. Solano Barliza, A. D. (2021). Revisión conceptual de sistemas de recomendación y geolocalización aplicados a la seguridad turística. Computer and Electronic Sciences: Theory and Applications, 2(2), 37–43. <https://doi.org/10.17981/CESTA.02.02.2021.05>
- [14]. Sommerville, I. (2015). Software Engineering (9th ed.).

Pearson Education Limited.