

Tipo de artículo: Artículo original

# Diseño de una arquitectura de software para el ecosistema tecnológico del Parque Científico-Tecnológico de La Habana

## *Design of a software architecture for the technological ecosystem of the Scientific-Technological Park of Havana*

Elizabeth Losa Pérez <sup>1\*</sup> , <https://orcid.org/0000-0001-9692-7174>

Camila Martínez Pita <sup>2</sup> , <https://orcid.org/0000-0003-2891-4448>

Febe Angel Ciudad Ricardo <sup>3</sup> , <https://orcid.org/0000-0002-0763-9005>

<sup>1</sup> Universidad de las Ciencias Informáticas. Carretera a San Antonio de los Baños, km 2 ½, Torrens, Boyeros, La Habana, CP. 19370. Correo electrónico: {elizabethlosa99, camipita21, fciudad}@gmail.com

\* Autor para correspondencia: [elizabethlosa99@gmail.com](mailto:elizabethlosa99@gmail.com)

### Resumen

El Parque Científico-Tecnológico de La Habana (PCT-LH) es una entidad de ciencia, tecnología e innovación (CTI) que se diseñó como un espacio para la ejecución de proyectos de investigación, desarrollo e innovación (I+D+i) en la rama de las Tecnologías de la Información y las Comunicaciones (TIC). La gran cantidad de procesos que existen dentro del PCT-LH resulta abrumadora tanto para los usuarios como para los encargados de su gestión y mantenimiento; por lo que se requiere de aplicaciones informáticas que contribuyan a solucionar problemas relacionados con su proceso de negocio. Esto hace necesario la implantación de un ecosistema tecnológico por cuanto, al estar todas las herramientas de cada área integradas, favorece la comunicación entre las mismas, permitiendo que los procesos internos sean más eficientes. Es por ello que surge la necesidad de diseñar una arquitectura de software para el ecosistema tecnológico del PCT-LH. Se realiza una fundamentación teórica sobre el tema en cuestión y se representa arquitectónicamente la propuesta de solución, efectuándose luego la validación de la misma mediante el criterio de un grupo de expertos en la temática.

**Palabras clave:** arquitectura de software, ciencia, ecosistema tecnológico, innovación, tecnología.

### Abstract

*The Scientific-Technological Park of Havana (PCT-LH) is a science, technology and innovation (STI) entity that was designed as a space for research, development and innovation (R+D+i) projects in the field of Information and Communication Technologies (ICT). The large number of processes that exist within the PCT-LH is overwhelming, both for users and for those responsible for its management and maintenance. Therefore, computer software is required to solve problems related to their business process. This makes it necessary to implement a technological ecosystem because, as all the tools of each area are integrated, it will favor the communication between them, allowing internal processes to be more efficient. That is why the need arises to design a software architecture for the technological ecosystem of the PCT-LH.)*

**Keywords:** software architecture, science, technological ecosystem, innovation, technology.

**Recibido:** 04/01/2023

**Aceptado:** 28/02/2023

**En línea:** 01/03/2023



Esta obra está bajo una licencia *Creative Commons* de tipo **Atribución 4.0 Internacional** (CC BY 4.0)

## Introducción

La investigación, el desarrollo y la innovación (I+D+i), son factores significativos para el crecimiento económico de un país o región. Así mismo, la obtención de productos y tecnologías nuevas y avanzadas son determinantes para su posición competitiva (Herrera-Márquez et al., 2015). Esta necesidad ha llevado a la concepción de modelos que permiten la interacción entre un conjunto de organizaciones, que, aplicando capacidades científicas, técnicas y sociales en un determinado contexto geográfico, realizan actividades orientadas a la generación, difusión, gestión y transferencia del conocimiento para crear productos y procesos innovadores. Este modelo de innovación ha llevado a la conformación de los Parques Científicos-Tecnológicos (PCT).

Como plantea (Rodríguez-Pose 2012) los Parques Científicos-Tecnológicos son una creación, generalmente pública, de una zona geográfica delimitada y destinada a favorecer el desarrollo y aplicación de actividades científicas y tecnológicas, teniendo por objetivo, la generación del crecimiento económico a largo plazo en el territorio en el que se instala el PCT.

Cuba ha apostado por el desarrollo de las Tecnologías de la Información y la Comunicación (TIC), que implica utilizar la información como elemento fundamental para generar valor y riqueza por medio de su transformación a conocimiento. En este sentido, un gran paso de avance fue la creación en febrero de 2020 del Parque Científico-Tecnológico de La Habana (PCT-LH) ubicado en los predios de la Universidad de las Ciencias Informáticas (UCI). El PCT-LH se diseñó como un espacio para la ejecución de proyectos de investigación, desarrollo e innovación en la rama de las TIC, que permita la obtención de resultados que impacten en el desarrollo del país (Torralbas y Delgado 2021).

La gran cantidad de procesos que existen dentro del PCT-LH resulta abrumadora tanto para los usuarios como para los encargados de su gestión y mantenimiento; por lo que se requiere de aplicaciones informáticas que contribuyan a solucionar problemas relacionados con su proceso de negocio, automatizar las tareas que realizan frecuentemente y que proporcionen asistencia a la hora de tomar decisiones. Para esto se hace necesario la construcción de eficientes sistemas informáticos, los que generalmente requieren la combinación de diferentes tecnologías y plataformas de hardware y software ofreciendo un alto nivel de rendimiento, adaptándose a las necesidades específicas de la institución y permitiendo la adición de nuevas funcionalidades con el menor esfuerzo posible (Alina et al. 2016).

A través de la integración de las aplicaciones que gestionan los procesos de diferentes áreas dentro del PCT-LH, se genera una base de datos donde se almacena la información relevante de la entidad y también indicadores de rendimiento de dichos procesos. Esto hace necesario la implantación de un ecosistema tecnológico por cuanto, al estar todas las herramientas de cada área integradas, favorece la comunicación entre las mismas, permitiendo que los



procesos internos sean más eficientes, lo que se traduce en oportunidades más rentables y un panorama general visiblemente más escalable.

Desde el punto de vista de (García-Peñalvo, 2018), un ecosistema tecnológico es una comunidad donde métodos educativos, políticas, reglamentos, aplicaciones y equipos de trabajo pueden coexistir de forma que sus procesos están interrelacionados y su aplicación se basa en los factores físicos del entorno tecnológico.

Un elemento de vital importancia en el diseño de un ecosistema tecnológico es la utilización de una adecuada y conveniente arquitectura de software, ya que constituye la base de cualquier sistema informático. Actualmente no existe una definición única para el concepto de arquitectura de software. El término ha sido abordado por un gran número de autores. No obstante, se reconoce como la definición más completa la dada por la (ISO/IEC/IEEE 42010) donde se plantea que «La arquitectura de software es la organización fundamental de un sistema enmarcada en sus componentes, las relaciones entre ellos, y el ambiente, y los principios que orientan su diseño y evolución». El diseño de una eficaz arquitectura de software, contrarresta los efectos negativos que pueden surgir durante el desarrollo de un software, ocupando un rol significativo en la estrategia de negocio de una organización que basa sus operaciones en el software.

Las condiciones que hacen necesarias la implantación de un ecosistema tecnológico para el PCT-LH, conlleva a que se defina como objetivo general diseñar la arquitectura de software del ecosistema tecnológico para el Parque Científico-Tecnológico de La Habana que favorezca la transformación digital de su proceso de gestión de la ciencia, la tecnología y la innovación.

## **Materiales y métodos**

Para el diseño de una arquitectura de software es necesario definir la metodología de desarrollo adecuada que sirva de guía para la descripción de la misma y permita reducir el nivel de dificultad, organizar las tareas, agilizar el proceso y mejorar el resultado final.

El Proceso Unificado de Desarrollo de Software (RUP, por sus siglas en inglés), es una metodología de desarrollo de software resultante de varios años de investigación y uso práctico, en la que se han unificado diferentes técnicas de producción de software. Esta se define como un conjunto de actividades necesarias para transformar los requisitos del usuario en un sistema de software (Ivar Jacobson, Grady Booch 2000). RUP presenta tres características fundamentales: dirigido por casos de uso, centrado en la arquitectura e Iterativo e incremental, y se divide en cuatro fases dentro de las cuales se definen un conjunto de objetivos específicos para cada fase.

Para el Proceso Unificado de Desarrollo, la arquitectura es un elemento primordial. Este abarca decisiones importantes sobre la organización del sistema, los elementos estructurales que corresponden al mismo, sus interfaces,



la composición de los elementos estructurales y el comportamiento en subsistemas progresivamente más grandes y como guía de esta organización, los estilos de la arquitectura.

La arquitectura en RUP se desarrolla mediante iteraciones, principalmente en la etapa de elaboración, donde el hito es la línea base de la arquitectura que no es más que donde se encuentran un conjunto de modelos que representan los casos de uso más importantes y sus realizaciones desde el punto de vista de la arquitectura (Ivar Jacobson, Grady Booch 2000). En RUP la arquitectura se representa mediante varias vistas que se centran en aspectos concretos y describen las principales partes del sistema.

Se selecciona el Proceso Unificado de Desarrollo (RUP) como metodología para la realización del diseño de la arquitectura de software precisamente por ser centrada en la arquitectura y que se desarrolla mediante iteraciones, comenzando por los casos de uso relevantes desde el punto de vista de la arquitectura.

### **Modelo 4+1 vistas**

La arquitectura como etapa de ingeniería y diseño orientada a objetos, describe cinco vistas clásicas del modelo “4+1 vistas” de (Kruchten 2006). Cada una de ellas (vista lógica, vista de procesos, vista física, vista de desarrollo, vista de casos de uso) se refiere a un conjunto de intereses de diferentes stakeholders del sistema. Para cada vista se definen un conjunto de elementos (componentes, contenedores y conectores), y en cada una de ellas se describen diagramas que usan su notación particular, aunque es un modelo bastante genérico y puede usar otra notación. Las distintas vistas no son completamente ortogonales o independientes, ya que los elementos de una vista están conectados a los elementos de las otras vistas siguiendo ciertas reglas de diseño.

Se emplea el enfoque arquitectónico basado en vistas “4+1 vistas” ya que es un modelo muy acertado para documentar la arquitectura de software del ecosistema tecnológico y así los stakeholders pueden entenderlo desde diferentes perspectivas. En el caso de la arquitectura propuesta, no es necesario la utilización de la vista de procesos, ya que esta arquitectura no cuenta con hilos ni procesos concurrentes. Se utilizaron las vistas de: casos de uso, lógica, procesos, despliegue e implementación. Esta selección corresponde a una óptima solución de la situación descrita para el desarrollo de la propuesta de solución.

### **Patrón Arquitectónico “En Capas”**

El patrón “en capas” es definido como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior (Garlan y Shaw 1994). En este patrón los componentes se estructuran en niveles o capas donde cada nivel invoca sólo al nivel inferior y las interfaces entre capas están claramente definidas. Contribuye a la disminución del acoplamiento, favorece la



portabilidad y la sustitución de componentes y proporciona un alto nivel de abstracción. Las capas que conforman este patrón son: Presentación, Negocio, Acceso a Datos y Base de Datos.

Se selecciona el patrón arquitectónico “en capas” ya que asegura una evolución sostenible del ecosistema tecnológico en todas sus vertientes, de tal forma que no afecte al sistema en producción, logrando un alto grado de integración y cohesión entre los componentes del ecosistema. Permite la gestión centralizada de los diferentes componentes del ecosistema, haciendo especial hincapié en la gestión de usuarios, con cobertura tanto para la gestión de los datos como de la autenticación en todo el ecosistema y brinda soporte para la toma de decisiones y para el análisis de los flujos de información que tienen lugar tanto dentro del ecosistema como provenientes del exterior y viceversa. Las capas utilizadas son: Presentación, Negocio, Acceso a datos e Infraestructura.

### **Estilo Arquitectónico “Arquitectura en Capas”**

El estilo arquitectónico “en capas” consta en dividir la aplicación en capas, con la intención de que cada capa tenga un rol muy definido, como podría ser, una capa de presentación (UI), una capa de reglas de negocio (servicios) y una capa de acceso a datos (DAO), sin embargo, este estilo arquitectónico no define cuantas capas debe tener la aplicación, sino más bien, se centra en la separación de la aplicación en capas (Iturralde 2020). En la práctica, la mayoría de las veces este estilo arquitectónico es implementado en 4 capas, presentación, negocio, persistencia y base de datos, sin embargo, es habitual ver que la capa de negocio y persistencia se combinan en una sola capa, sobre todo cuando la lógica de persistencia está incrustada dentro de la capa de negocio.

Se decide utilizar el estilo “en capas” mediante las capas de: Presentación, Negocio, Acceso a datos e Infraestructura, puesto que tiene como ventaja testear cada capa individualmente debido a la separación clara de responsabilidades que existe entre ellas y, al hacer un cambio, si se implementó correctamente cada responsabilidad, este cambio solo impacta a la capa responsable y no a todas, ventaja que será de gran beneficio para el diseño de la propuesta de solución.

### **Método Delphi**

En el proceso de desarrollo del software ocurren diferentes momentos de validación, pero en esencia siempre se comienza con la validación de las especificaciones del software (requisitos, modelo, arquitectura), después se valida la implementación del mismo (código fuente) y por último el despliegue de la aplicación (interacción con otros sistemas). En este caso sólo se realizará la validación de las especificaciones de la arquitectura de software propuesta mediante el método Delphi para obtener los resultados que demuestren la validez de la arquitectura de software del ecosistema tecnológico para el Parque Científico-Tecnológico de La Habana.



## Resultados y discusión

La arquitectura de software del ecosistema tecnológico para el Parque Científico-Tecnológico de La Habana queda definida en cuatro capas: Infraestructura, Acceso a Datos, Negocio y Presentación.

En primer lugar, la capa Infraestructura proporciona todos los servicios necesarios a nivel interno para que el sistema funcione correctamente. En esta capa deben existir, como mínimo, tres componentes: el servidor de correo; la base de datos que abarca tanto la gestión de usuarios como la autenticación; y la monitorización de los flujos de información resultantes de los procesos fundamentales del ecosistema tecnológico.

La capa Acceso a Datos no separa la capa Negocio de la capa Infraestructura. Los ecosistemas tecnológicos gestionan gran cantidad de información, desde la información generada a partir de la interacción de los usuarios con el sistema hasta la información proveniente del exterior y que pasa a formar parte de alguno de los componentes del ecosistema. Una de las características que diferencian la arquitectura de un ecosistema de la arquitectura tradicional por capas es que toda esa información no se centraliza. Cada componente gestiona su propia información e interactúa con otros componentes mediante flujos predefinidos, pero hay un conjunto de datos que deben estar accesibles a todos los componentes del ecosistema, de tal forma que todos aquellos datos que se desea tener a disposición de todo el ecosistema se gestionarán de manera centralizada.

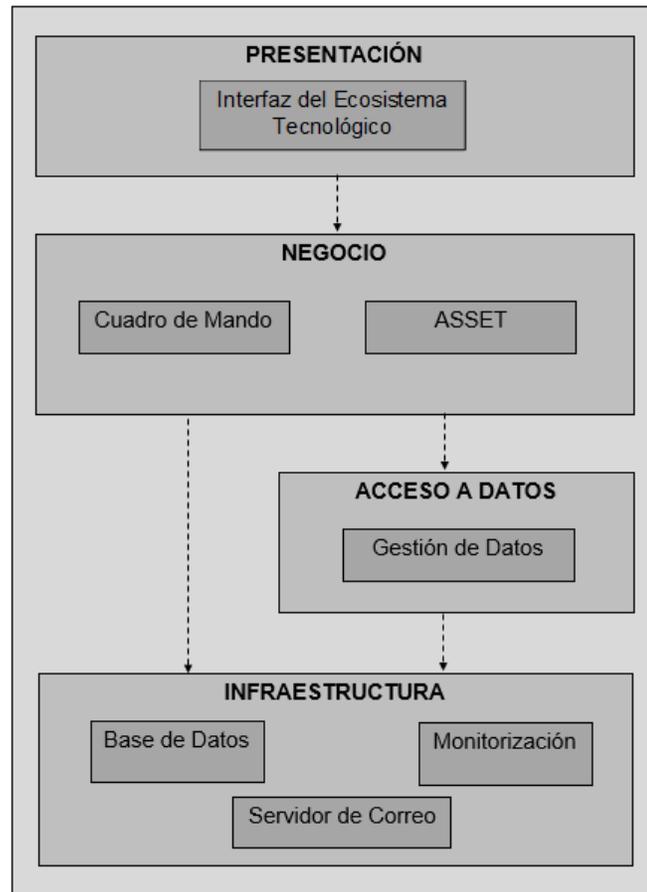
Por este motivo se ha introducido en el patrón arquitectónico la capa Acceso a Datos. Los diferentes componentes del ecosistema que necesiten datos accederán a un único componente de gestión de datos, lo cual facilita enormemente la gestión de los mismos. La capa no es una capa al uso ya que pueden existir procesos que no utilicen este tipo de datos, por este motivo se representa con una longitud menor.

La tercera capa Negocio engloba el conjunto de procesos con sus respectivas herramientas de gestión que se llevan a cabo en el PCT-LH y que proporcionan los diferentes servicios a nivel de usuario. Estos procesos son: Incubación de Proyectos, Incubación de Entidad de Base Tecnológica y Gestión Económica-Financiera. Las herramientas de gestión utilizadas son: Cuadro de Mando y Asset.

La última capa se centra únicamente en la presentación. El ecosistema debe transmitir unicidad y para ello es necesario que la presentación del mismo no esté fuertemente ligada a cada uno de sus componentes. Esta capa también debe encargarse de asegurar la accesibilidad del ecosistema desde cualquier tipo de dispositivos.

La siguiente figura muestra la representación arquitectónica en cuatro capas del ecosistema tecnológico para el Parque Científico-Tecnológico de La Habana:





**Figura 1.** Representación gráfica de la Arquitectura 4 Capas del Ecosistema Tecnológico para el PCT-LH.

## Validación

Se aplicó una primera encuesta de autovaloración a 10 candidatos, de los cuales, 7 son especialistas, 4 de ellos con el rol de Arquitecto de Software certificado y 3 que ejercen actualmente este rol en el Centro de Gobierno Electrónico (CEGEL) y en el Centro de Informatización de la Gestión Documental (CIGED) en la UCI. El resto de los expertos son 3 profesores que se han desempeñado como arquitectos de softwares en proyectos donde han estado vinculados.

Para la selección de los expertos se emplea la valoración por competencias mediante una encuesta de autovaloración. Este método consiste en calcular el Coeficiente de competencia (K) a partir del Coeficiente de conocimiento sobre el tema (Kc) y el Coeficiente de argumentación (Ka) mediante la siguiente fórmula:  $K = \frac{1}{2} (Kc + Ka)$ .

El resultado del Coeficiente de competencias de todos los encuestados se muestra en la tabla siguiente:



**Tabla 1:** Coeficiente de competencias de los encuestados.

Número de expertos	Kc	Ka	K	Interpretación
1	0.6	0.9	0.75	Medio
2	0.9	1.0	0.95	Alto
3	0.9	1.0	0.95	Alto
4	0.9	1.0	0.95	Alto
5	1.0	1.0	1.0	Alto
6	1.0	1.0	1.0	Alto
7	1.0	0.9	0.95	Alto
8	0.8	0.9	0.85	Alto
9	0.8	0.9	0.85	Alto
10	1.0	0.9	0.95	Alto

De los 10 candidatos encuestados, 9 tienen un Coeficiente de competencia Alto y 1 Medio, siendo de 0.92 el promedio del Coeficiente de competencia. Por lo cual se tomó la decisión de que los 10 candidatos sean incluidos en el grupo de expertos para la evaluación de la propuesta de solución.

Los expertos recibieron un resumen de la arquitectura de software planteada. Posteriormente se les realizó una encuesta para evaluar el grado de factibilidad de los elementos que conforman la propuesta de solución, mediante la definición de indicadores claros, entendibles, sencillos y precisos para respaldar el diseño de la misma. En el análisis y procesamiento de los resultados se toman los valores de los criterios de adecuación (Muy Adecuado (MA), Bastante Adecuado (BA), Adecuado (A), Poco Adecuado (PA), No Adecuado (NA)) de cada uno de los expertos sobre la validación de la propuesta, agrupándose por indicadores en una tabla de frecuencias acumuladas.

**Tabla 2** Tabla de frecuencias acumuladas.

Indicadores	MA	BA	A	PA	NA	Total
1	5	5	0	0	0	10
2	3	4	2	1	0	10
3	2	5	3	0	0	10
4	1	6	3	0	0	10
5	3	3	4	0	0	10
6	2	7	1	0	0	10



Tabulados los datos, se realizan los cálculos de: frecuencia absoluta acumulada, frecuencia relativa acumulada e inversa de la distribución normal estándar acumulativa:

**Tabla 3:** Tabla de frecuencias absolutas acumuladas.

Indicadores	MA	BA	A	PA	NA
1	5	10	10	10	10
2	3	7	9	10	10
3	2	7	10	10	10
4	1	7	10	10	10
5	3	6	10	10	10
6	2	9	10	10	10

Se elimina el último criterio de adecuación (NA) ya que sólo se necesitan cuatro puntos de corte.

**Tabla 4** Tabla de frecuencias relativas acumuladas

Indicadores	MA	BA	A	PA
1	0.5	1.0	1.0	1.0
2	0.3	0.7	0.9	1.0
3	0.2	.0.7	1.0	1.0
4	0.1	0.7	1.0	1.0
5	0.3	0.6	1.0	1.0
6	0.2	0.9	1.0	1.0

Aplicando la función “Dist. Normal. Standard Inv” en la herramienta Microsoft Excel, se obtiene la inversa de la distribución normal estándar acumulativa. A esta tabla se le adicionan 4 columnas y 2 filas para mostrar los resultados de:

- La suma de las columnas (Suma C).
- La suma de filas (Suma F).
- El promedio de las columnas (P).
- El promedio de las filas (Puntos de corte).



- El valor de N: División entre la sumatoria de filas y columnas y el resultado de multiplicar el número de indicadores (5) por el número de categorías (5).
- El valor N-P: Promedio que otorgan los expertos para cada indicador propuesto.

Lo anteriormente explicado se resume en la siguiente tabla:

**Tabla 5:** Tabla de puntos de corte.

N=1.50								
Indicadores	MA	BA	A	PA	Suma C	P	N-P	Grado de adecuación
1	0	3,72	3,72	3,72	11,16	2.79	-1.29	<b>Muy Adecuado</b>
2	-0,52	0,52	1,28	3,72	5	1.25	0.25	<b>Bastante Adecuado</b>
3	-0,84	0,52	3,72	3,72	7,12	1.78	-0.28	<b>Bastante Adecuado</b>
4	-1,28	0,52	3,72	3,72	6,68	1.67	-0.17	<b>Bastante Adecuado</b>
5	-0,52	0,25	3,72	3,72	7,17	1.79	-0.29	<b>Bastante Adecuado</b>
6	-0,84	1,28	3,72	3,72	7,88	1.97	-0.47	<b>Bastante Adecuado</b>
<b>Suma F</b>	-4	6.81	19.88	22.32	<b>45.01</b>			
<b>Puntos de corte</b>	<b>-0.66</b>	<b>1.135</b>	<b>3.31</b>	<b>3.72</b>				

Se utilizan los puntos de corte para determinar la categoría de cada criterio según la opinión de los expertos consultados. Con ellos se seleccionó el grado de adecuación teniendo en cuenta los siguientes valores:

**Tabla 6:** Categoría de cada criterio.

Muy adecuado	Bastante adecuado	Adecuado	Poco adecuado	No adecuado
N-P < -0.66	-0.66 < N-P < 1.135	1.135 < N-P < 3.31	3.31 < N-P < 3.72	N-P > 3.72



Con la aplicación del método Delphi, se valida la propuesta arquitectónica, obteniendo resultados satisfactorios según el criterio de los expertos consultados, ya que el indicador “objetivos que persigue la propuesta” es muy adecuado y el resto de los indicadores son bastante adecuados.

## Conclusiones

Con el diseño de la arquitectura de software para el ecosistema tecnológico del Parque Científico-Tecnológico de La Habana, se alcanza un gran paso de avance, ya que sirve de guía para futuras arquitecturas de softwares con este fin y contribuye a la integración e informatización de cada uno de los procesos que se llevan a cabo en este para su mejor desarrollo por parte de los usuarios y de los encargados de su mantenimiento. La validación de la propuesta de solución mediante el criterio de un grupo de expertos que estudiaron un resumen de la investigación, aseguró la obtención de un resultado positivo a todo aquel equipo de desarrollo que aplique la propuesta arquitectónica.

## Conflictos de intereses

Los autores no poseen conflictos de intereses.

## Contribución de los autores

1. Conceptualización: Elizabeth Losa Pérez, Camila Martínez Pita, Febe Angel Ciudad Ricardo
2. Curación de datos: Elizabeth Losa Pérez, Camila Martínez Pita
3. Análisis formal: Camila Martínez Pita, Febe Angel Ciudad Ricardo, Febe Angel Ciudad Ricardo
4. Investigación: Elizabeth Losa Pérez, Camila Martínez Pita
5. Metodología: Elizabeth Losa Pérez, Febe Angel Ciudad Ricardo
6. Software: Elizabeth Losa Pérez, Camila Martínez Pita
7. Validación: Elizabeth Losa Pérez, Camila Martínez Pita, Febe Angel Ciudad Ricardo
8. Visualización: Elizabeth Losa Pérez, Camila Martínez Pita
9. Redacción – borrador original: Elizabeth Losa Pérez, Camila Martínez Pita, Febe Angel Ciudad Ricardo
10. Redacción – revisión y edición: Elizabeth Losa Pérez, Camila Martínez Pita, Febe Angel Ciudad Ricardo



## Financiamiento

La investigación no requirió fuente de financiamiento externa.

## Referencias

- ALINA, I., RODRÍGUEZ, D., LUIS, I.I., SILVA, G. y II, R., 2016. Arquitectura de software para el sistema de visualización médica Vismedic Software Architecture for the Vismedic medical visualization system. , vol. 8, no. 1, pp. 75-86.
- GARCÍA-PEÑALVO, F.J., 2018. Ecosistemas Tecnológicos Universitarios. , pp. 164-170.
- GARLAN, D. y SHAW, M., 1994. An Introduction to Software Architecture. , no. January.
- HERRERA-MÁRQUEZ, J.J., SALAS-NAVARRO, L.C., DOMÍNGUEZ-MORÉ, G.P. y TORRES-SAUMETH, K.M., 2015. Situación del Caribe colombiano \*. , vol. 11, no. 2, pp. 112-130.
- ISO/IEC/IEEE 42010: Defining «architecture». [en línea], [sin fecha]. Disponible en: <http://www.iso-architecture.org/ieee-1471/defining-architecture.html>.
- ITURRALDE, O.J.B., 2020. *Introducción a la arquitectura de software – Un enfoque práctico*. 1ra. S.l.: s.n.
- IVAR JACOBSON, GRADY BOOCH, J.R., 2000. *El proceso unificado de desarrollo de software*. S.l.: s.n.
- KRUCHTEN, P., 2006. Planos Arquitectónicos : El Modelo de “ 4 + 1 ” Vistas de la La Arquitectura L ´ , vol. 12, no. 6, pp. 1-16.
- RODRÍGUEZ-POSE, A., 2012. Los parques científicos y tecnológicos en América Latina. ,
- TORRALBAS, R. y DELGADO, M., 2021. Creación, Organización Y Gestión Del Parque Científico Tecnológico De La Habana. *Gastronomía ecuatoriana y turismo local*. [en línea], vol. 1, no. 69, pp. 5-24. Disponible en: <http://scielo.sld.cu/pdf/rus/v13n1/2218-3620-rus-13-01-346.pdf>.

