

Software para autoevaluación de programas académicos en instituciones de educación superior

Software for self-evaluation of educational programs in colleges and universities

Alvaro Hernán Alarcón López¹, José Miguel Llanos Mosquera², Julián Andrés Quimbayo Castro³, Eliana Andrea Londoño Vieda⁴
y Tomas Octavio Rodríguez Herrera⁵

^{1,2,3,4,5} Facultad de Ingeniería, Corporación Universitaria del Huila - CORHUILA Neiva, Colombia.

*Autor de correspondencia: alvaro.alarcon@corhuila.edu.co

RESUMEN– En la actualidad las instituciones de educación superior en Colombia tienen como uno de sus principales objetivos ser reconocidas por sus altos estándares de calidad, para ello buscan ser certificadas por organismos de regulación mediante la acreditación institucional y de programas académicos. Uno de los procesos a desarrollar para alcanzar esta distinción es la autoevaluación, por medio de esta actividad las instituciones de educación superior efectúan un control interno; esta actividad por lo general se realiza sin el apoyo de software por lo cual se deben destinar una gran cantidad de horas de trabajo de docentes y personal administrativo para la realización de la misma, esta situación puede generar retrasos en la generación del plan de mejoramiento y en la toma de decisiones para solucionar las falencias encontradas. Sin embargo, algunas instituciones de educación superior han optado por implementar software para realizar estos procesos internos, siendo positivo el resultado de esta intervención tecnológica. El presente proyecto se enfocó en el desarrollo de autoevaluación para la Corporación Universitaria del Huila; los requerimientos fueron obtenidos a partir de entrevistas a expertos y de los lineamientos del consejo nacional de acreditación. Para el diseño del software se utilizó el modelo de vistas de arquitectura 4+1 y el patrón arquitectónico Modelo Vista Controlador – MVC. 30 usuarios realizaron pruebas de funcionamiento al software, en estas se pudo constatar un alto grado de satisfacción en cuanto al propósito, rendimiento y usabilidad de la aplicación web.

Palabras clave– Acreditación, aplicaciones web, autoevaluación, calidad, diseño, software.

ABSTRACT– At present, higher education institutions in Colombia have as one of their main objectives to be recognized for their high quality standards, for that purpose they seek to be certified by regulatory bodies through institutional accreditation and academic programs. One of the processes to develop to achieve this distinction is self-evaluation, through this activity higher education institutions make an internal control, this activity is usually done without the support of software so they must allocate a large number of hours of work of teachers and administrative staff to carry it out, this situation can generate delays in the generation of the improvement plan and decision making to solve the shortcomings found. However, some higher education institutions have opted to implement software to carry out these internal processes, the result of this technological intervention being positive. The present project focused on the development of self-evaluation for the Corporación Universitaria del Huila; the requirements were obtained from interviews with experts and from the guidelines of the national accreditation council. For the design of the software, we used the 4+1 architectural views model and the architectural pattern Model View Controller - MVC. The software was tested by 30 users, who were highly satisfied with the purpose, performance and usability of the web application.

Keywords– Accreditation, web applications, autoevaluation, quality, design, software.

1. Introducción

El proceso de autoevaluación de programas de pregrado en instituciones de educación superior en Colombia; consiste en una observación crítica de los procesos internos por parte de estudiantes, docentes, personal administrativo, egresados y sector externo, en busca de generar planes de mejoramiento que posibiliten el mejoramiento continuo [1] [2].

Hoy en día se ha generalizado el uso de aplicaciones web, como apoyo para el desarrollo de este tipo de actividades educativas y administrativas en instituciones de educación superior; situación que ha propiciado mejoras en los procesos realizados al interior de estas.

Como ejemplo de lo descrito, en Colombia algunas universidades han optado por la incorporación de software, como soporte del proceso de autoevaluación con fines de acreditación de sus programas académicos de pregrado [3] [4].

Actualmente la Corporación Universitaria del Huila – CORHUILA, desarrolla sus procesos de autoevaluación con fines de acreditación institucional y de programas académicos sin el apoyo de software. Esta situación implica que sea necesario la presencia de personal administrativo y docente para la aplicación de encuestas, la tabulación y el análisis estadístico de las mismas; siendo necesarias varias horas de trabajo para la realización de estas actividades, tiempo que podría ser empleado para la ejecución de otras labores académicas al interior de los programas y facultades de la institución.

Esta dinámica de desarrollo del proceso de autoevaluación implica la inversión de varias horas de trabajo para lograr establecer un plan de mejoramiento; situación que eventualmente podría perjudicar la oportuna toma de decisiones por parte de las directivas de la institución, con el propósito de dar solución a las falencias encontradas.

Por tanto, teniendo en cuenta que la implementación de software en el proceso de autoevaluación en otras instituciones de educación, ha permitido alcanzar una mayor eficiencia en esta actividad de acreditación [5]; se logró establecer la hipótesis que el uso de una aplicación web, podría ayudar a optimizar el desarrollo del mencionado proceso en los programas académicos de CORHUILA.

De esta forma el presente trabajo presenta el desarrollo de un software para gestionar el proceso de autoevaluación con fines de acreditación en el programa de Ingeniería de Sistemas de la Corporación Universitaria del Huila - CORHUILA, institución de educación superior ubicada en la ciudad de Neiva departamento del Huila – Colombia.

2. Fundamentación Teórica y Estado del Arte

En primer lugar cabe destacar que para Colombia la acreditación de los programas académicos, es voluntaria para las instituciones de educación superior y tiene el objetivo de que estas sean reconocidas por sus altos niveles de calidad [6].

Además, la autoevaluación es uno de los elementos fundamentales para el proceso de acreditación de una institución de educación superior y sus programas académicos; consiste en la evaluación interna por parte de docentes, administrativos, estudiantes y egresados, con el propósito de determinar las fortalezas, debilidades

y oportunidades de mejora. De esta forma al final del proceso se realiza el diseño y ejecución de un plan de mejoramiento con el propósito de asegurar la calidad de los servicios educativos [7].

Así mismo la calidad de los programas académicos, está relacionada con las condiciones en las cuales funciona una institución de educación superior; está definida por los procesos, los actores, los resultados académicos, la infraestructura y demás situaciones educativas al interior de las universidades [8].

Estas condiciones de alta calidad de programas académicos son definidas y evaluadas por el consejo Nacional de acreditación. Este es un organismo vinculado al Ministerio de Educación Nacional, el cual tiene el propósito de coordinar, planificar, recomendar y asesorar a las instituciones de educación superior colombianas, en temas relacionados con la acreditación de alta calidad [9].

De otro lado en la actualidad el uso de la tecnología ha generado un gran impacto en la sociedad; de tal forma que las empresas, las personas e instituciones han hecho uso de algún tipo software para el desarrollo de sus procesos. Estos por lo general tienen la posibilidad de actualizarse, agregar nuevas funciones y facilitar la realización de las actividades laborales [10]. Algunos tienen funciones específicas para facilitar las operaciones de negocios y la respectiva toma de decisiones en tiempo real [11]. Existen además plataformas para el desarrollo de software IDE, un ejemplo de ellas es ASP.Net, la cual pertenece a Microsoft y permite el desarrollo de aplicaciones web a través de un lenguaje integrado en la .Net framework [12].

Se han desarrollado algunos estudios y proyectos de software, para procesos de autoevaluación y/o acreditación de alta calidad en instituciones de educación superior. Uno de estos fue realizado en la Universidad Libre – seccional Bogotá; la aplicación denominada SAAACI pretendía apoyar el proceso de la acreditación de alta calidad de la institución, ya que permitía obtener los resultados de las autoevaluaciones realizadas, para posteriormente crear del plan de mejoramiento [3].

Otro ejemplo es el software denominado SAP-UdeC v1.0, este se desarrolló en la Universidad de Cartagena como plataforma web para la autoevaluación de programas, con fines de Acreditación o con fines de renovación de registro calificado [4].

En este mismo sentido el software SAEPRO V1.0, fue desarrollado para apoyar el proceso de autoevaluación de programas de pregrado; este contaba

con aplicación web y de escritorio, con el propósito de proporcionar facilidades para la construcción de instrumentos, recolección y procesamiento de información y análisis de resultados. Además, la aplicación Web permitía recolectar la información de los egresados y empleadores, evitando la aplicación de encuestas físicas a los mismos [5]. Sin embargo a nivel regional, en el departamento del Huila no se encontró evidencia documental de desarrollos de software de autoevaluación con fines de acreditación para los programas académicos de las instituciones de educación superior.

3. Metodología

Para el desarrollo de la investigación se tuvo en cuenta dos momentos importantes, uno, la metodología de tipo descriptiva la cual se basa en determinar las características, situaciones, costumbres y actitudes a través de la descripción de procesos, actividades u objetos dentro de una población [13]; dos, la metodología de desarrollo ágil scrum [14] la cual permite por medio de sprint o iteraciones realizar entregas tempranas en un porcentaje mayor o igual a un 80% de funcionalidad según los requerimientos del cliente.

Teniendo en cuenta las dos líneas de trabajo complementario se procede a determinar la muestra de tipo no probabilístico [15], esto quiere decir que dicho proceso no brinda a los individuos de la población iguales condiciones de ser seleccionados, siendo aún más preciso poder establecer características específicas del grupo seleccionado. En el año 2017 se tomó como referencia el 10% de los estudiantes activos matriculados adscritos al programa de ingeniería de sistemas siendo 38 el número de escogidos de un total de 380, para realizar las pruebas al software se tuvo en cuenta que debían tener como criterios de exclusión haber realizado de forma manual la encuesta que se realiza en el proceso de autoevaluación por parte de la Corporación Universitaria del Huila – Corhuila [16].

Por tanto, Se utilizaron dos grupos, uno de ellos fue el de posprueba quienes realizaron las encuestas necesarias derivadas del proceso de autoevaluación de forma manual, el otro corresponde al grupo de control a quienes se les aplicaron las mismas encuestas, pero usando el software desarrollado.

Además, se trabajó con un enfoque cualitativo en la etapa de recolección de información debido a que el propósito fue el de conocer la manera en que los actores

del proceso de autoevaluación experimentan el funcionamiento del programa y establecen sus puntos de vista. Además se usó el enfoque cuantitativo para la obtención de resultados y poner a prueba la hipótesis planteada por medio de una medición numérica y un análisis estadístico [16].

En la investigación se desarrollaron varias fases, dentro de las cuales se encuentran requerimientos, diseño, desarrollo y pruebas enfocadas al software. También se utilizaron las fases de recolección y análisis de la información para la generación de resultados.

3.1 Fase 1: Requisitos

Para el levantamiento de requisitos se utilizó el estándar IEEE 29148-2011 [17] Ingeniería de sistemas y software - Procesos del ciclo de vida --Ingeniería de requisitos. En el documento se definió el alcance del proyecto, los stakeholders del proyecto, los actores, los requisitos funcionales y no funcionales del software de autoevaluación. Los requisitos funcionales más relevantes fueron: Gestión de usuarios, realización de encuestas, generación de ponderación para las características definidas en los factores de acreditación [6] y generación de los reportes. Los requisitos no funcionales se definieron con base a los atributos de calidad de usabilidad, seguridad, y rendimiento.

3.2 Fase 2: Diseño del Software

Para el diseño del software se utilizó el modelo de vistas de arquitectura 4+1 definido por Philippe Kruchten, que consiste en la generación de varios diagramas UML para la documentación del proyecto. Dentro de las vista encontramos la de escenarios donde se generan los diagramas de casos de uso a partir de los requisitos funcionales definidos en la fase 1; la vista lógica donde se generan los diagramas de clases, comunicación y secuencia que se utilizarán en la fase de desarrollo; la vista de procesos donde se generan los diagramas de actividad para identificar los actores y procesos que realizan; la vista de despliegue donde se generan los diagrama de componentes y paquetes para la implementación del software y finalmente la vista física donde se genera el diagrama de despliegue de la solución [18].



Figura 1. Modelo de vistas de arquitectura 4+1 de Philippe Kruchten. Fuente: <https://jarroba.com/modelo-41-vistas-de-kruchten-para-dummies/>

3.3 Fase 3: Desarrollo del Software

Para el desarrollo del Software se definió la metodología Scrum [19], porque permite realizar entregas continuas de las funcionalidades del software al cliente, validar los requisitos y obtener retroalimentación de todas las personas involucradas en el proyecto a medida que se desarrolla el producto [20].

El software se desarrolló con el patrón arquitectónico Modelo Vista Controlador – MVC, que consiste en tres componentes. La Vista se encarga de interactuar con el usuario y enviar las solicitudes que él genera al Controlador para que las gestione con el Modelo. El Modelo se encarga de pasar dichas solicitudes al motor de base de datos y esperar su respuesta para pasárselas al Controlador. Finalmente, el Controlador toma las respuestas generadas por el Modelo y se las pasa a la Vista para que se las presente al usuario a través de un navegador web.

La herramienta de desarrollo utilizada para el software fue Visual Studio, el lenguaje de programación C Sharp y el motor de base de datos SQL Server. Las tecnologías utilizadas fueron bootstrap para la vista; para el controlador y el modelo se utilizó C Sharp con entity framework. Finalmente se implementó y desplegó la aplicación web en el servidor Internet Information Server – IIS.



Figura 2. Página de inicio del software

Fuente propia



Figura 3. Ingreso de usuarios al sistema. Fuente propia

En la figura 2 se presenta el módulo de ingreso de usuarios al sistema.

3.4 Fase 4: Pruebas

Se realizaron pruebas unitarias al software con la fin de comprobar que las funcionalidades de cada módulo cumplieran con los objetivos propuestos; para ello se aplicaron casos de pruebas, estos se realizaron teniendo en cuenta una secuencia de acciones, pasos y resultados esperados [21]. También se aplicaron pruebas de usabilidad al software, con el fin de determinar la experiencia del usuario al interactuar con la solución para determinar posibles mejoras [22].

3.5 Fase 5: Recolección de Información

En este proceso se manejaron dos etapas, la primera que corresponde a la realización de entrevistas para conocer las impresiones acerca de la funcionalidad y usabilidad del software acorde a la normatividad vigente en el momento.

En la primera etapa se optó por la realización de entrevistas al comité encargado del proceso de acreditación del programa académico de ingeniería de sistemas de la Corporación Universitaria del Huila – CORHUILA. Esta actividad se efectuó con el propósito de entender cómo se desarrollan las actividades del proceso de autoevaluación, en el programa académico de ingeniería de sistemas.

En la segunda etapa se realizaron encuestas a estudiantes, esto teniendo en cuenta que solo se le aplicó el instrumento a estudiantes que hubiesen diligenciado las encuestas de forma manual; por tanto los estudiantes seleccionados se encontraron en el rango de tercer a décimo semestre.

3.6 Fase 6: Análisis de Información

El análisis de las entrevistas realizadas permitió generar las siguientes apreciaciones:

- Se determinó que los procesos actuales en el desarrollo de la autoevaluación poseen un alto consumo de tiempo tanto en la aplicación de las encuestas como en los procesos de tabulación y ponderación a estas, que hacen que los directores de programa y docentes involucrados pierdan capacidad productiva en sus diversas actividades diarias.
- Inconvenientes al no tener contacto con todos los actores que intervienen en este proceso en especial los egresados, ya que son una población que requiere mayor seguimiento y contacto constante como garantes del proceso de enseñanza.
- Impacto negativo en el medio ambiente por el alto consumo de papel.

4. Resultados

Se aplicaron encuestas a 30 estudiantes del programa de Ingeniería de Sistemas de CORHUILA, los cuales habían realizado pruebas de funcionamiento al software. Las preguntas realizadas se plantearon teniendo en cuenta los criterios establecidos en la ISO 25010 del 2011 [23] y la teoría de calidad de uso del software. Además para medir los resultados se utilizó la escala de Likert, ya que esta permite medir actitudes, por medio de ítems presentados en forma de afirmaciones o juicios y observar la reacción de los sujetos [24].

Algunas de las preguntas evaluadas en la encuesta realizada fueron las siguientes:

¿En qué medida cree usted que el software cumple con el objetivo propuesto?

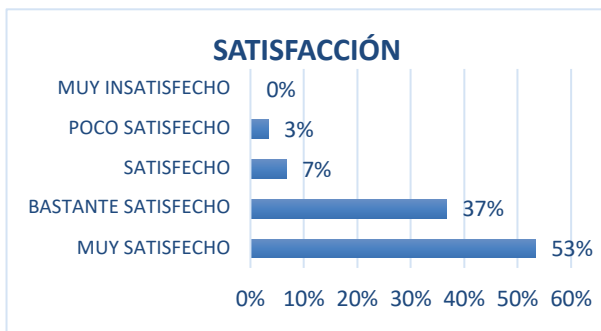


Figura 4. Cumplimiento de objetivo propuesto. Fuente propia

Teniendo en cuenta estos datos podemos inferir que el ítem con mayor porcentaje de respuestas fue muy satisfecho que corresponde a un porcentaje del 53%, por tanto el software logra el objetivo propuesto y cumple con las expectativas del usuario, además el porcentaje de encuestados insatisfechos es muy bajo e inferior al 10% del total de alumnos. Lo anterior indica según la gran mayoría de estudiantes (97%), que el software es apropiado para desarrollar el proceso de autoevaluación en el programa de Ingeniería de Sistemas.

¿Cuál es su nivel de satisfacción respecto al rendimiento del software?

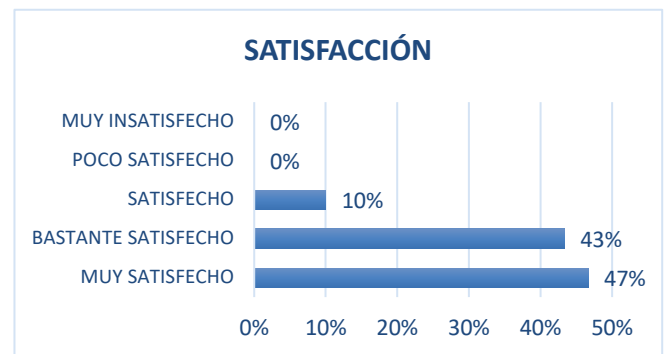


Figura 5. Rendimiento del software. Fuente propia

En este ítem se puede observar, que de los 30 estudiantes encuestados, 3 de ellos (10%) indican que se encuentran satisfechos respecto al rendimiento del software, 13 alumnos (43%) afirman que están bastante satisfechos, 14 educandos (47%) están muy satisfechos. Por tanto, podemos inferir que el 90% de encuestados se encuentran bastante satisfechos y muy satisfechos respecto al rendimiento del software, ya que se encuentran ubicados dentro de las alternativas que presentan mayor valor, además no se presentaron respuestas en las alternativas de menor valor en este caso poco satisfecho y muy insatisfecho. De los datos anteriores se puede concluir que el total de los encuestados piensa que el software presenta un rendimiento adecuado para desarrollar actividades de autoevaluación.

¿Indique el nivel de facilidad con el que puede conocer de forma rápida y clara que acciones realizar en la aplicación?

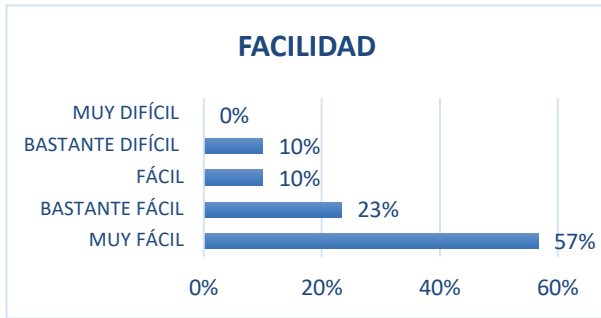


Figura 6. Uso de la aplicación. Fuente propia

De los 30 encuestados, 3 personas (10%) indican que es bastante difícil conocer de forma rápida y clara que acciones realizar en la aplicación, 3 personas (10%) que es fácil, 7 personas (23%) que es bastante fácil y 17 personas (57%) que es muy fácil. Respecto a lo anterior el ítem que predomina es muy fácil, superando en un 50% a los demás ítem, por tanto en su gran mayoría la población encuestada considera que es muy fácil conocer de forma rápida y clara que acciones realizar en la aplicación, por tanto se infiere que el software cumple con su objetivo. De los datos anteriores se debe tener en cuenta que para un 10% de la población encuestada, la aplicación presenta problemas de usabilidad, por tanto se deben realizar ajustes, con el propósito de poder realizar la implementación de esta.

5. Conclusiones

Como resultado de la investigación se concluye que es de gran utilidad contar con un software para realizar los procesos de autoevaluación de la Corporación Universitaria del Huila-Corhuila, esto teniendo en cuenta los resultados favorables de las encuestas realizadas.

Los requisitos funcionales y no funcionales fueron la base del desarrollo del software, ya que a partir de ellos se establecieron las respectivas funcionalidades de cada módulo de este.

De acuerdo con la revisión realizada se pudo comprobar que no se ha realizado ninguna investigación respecto a este tipo de software en el departamento del Huila, por tanto se está generando un aporte a nivel regional, esto en cuanto al desarrollo de esta investigación y del software para apoyar el proceso de autoevaluación con fines de acreditación de los programas académicos.

De acuerdo con los resultados obtenidos mediante la aplicación de las 30 encuestas realizadas respecto a las 11 preguntas formuladas, se evidencia que los

encuestados se encuentran muy satisfechos respecto al software y su utilidad.

Otra de las contribuciones que se presentan con el software es facilitar el proceso de ponderación, y generación de reportes, que es esencial para los encargados del proceso de acreditación de los programas académicos.

Para la realización de las pruebas unitarias se realizaron casos de prueba, estos están asociados a los casos de uso y a su vez a los requerimientos funcionales y no funcionales, en estos casos de prueba se detallaron paso a paso las funcionalidades y los resultados esperados, esto con la finalidad de verificar que el software cumpla la funcionalidad prevista en cada módulo.

6. Agradecimiento

“El desarrollo de este trabajo fue financiado por la Corporación Universitaria del Huila -CORHUILA”.

7. Referencias

- [1] Universidad de San Buenaventura, “Guía para la Autoevaluación de Programas Académicos.” Cali, p. 72, 2016.
- [2] Universidad de Cartagena, “Modelo de autoevaluación de programas.” Cartagena, p. 119, 2012.
- [3] M. Andrea, C. Villalba, J. Andres, and H. Gonzalez, “Sistema de apoyo de acreditación de alta calidad institucional SAAACI para la universidad libre – seccional Bogotá,” 2012.
- [4] M. E. Monroy Ríos, E. M. Gómez Bustamante, O. J. Ballesteros Pacheco, and A. Gonzales Villamizar, “Sap Udec sistema de autoevaluación de programas.” Universidad de Cartagena, Cartagena, 2013.
- [5] A. Cabarcas Alvarez, D. De las Aguas Ramirez, and R. A. Mendoza Garrido, “SAEPRO v1.0: Software de Autoevaluación de Programas. Una experiencia exitosa de transferencia de tecnología en Instituciones de Educación Superior,” vol. 5, no. 1, 2009.
- [6] D. M. Ramírez *et al.*, “Lineamientos para la acreditación de programas de pregrado,” Consejo Nacional de acreditación, 2013.
- [7] A. Campo *et al.*, “GUIA DE PROCEDIMIENTO -CNA 03- CUARTA EDICIÓN,” 2006.

- [8] “Acreditación de programas pregrado - CNA,” 2018. [Online]. Available: <https://www.cna.gov.co/1741/article-186377.html>. [Accessed: 24-Apr-2018].
- [9] “Consejo Nacional de Acreditación - CNA.” [Online]. Available: <https://www.cna.gov.co/1741/article-186382.html>. [Accessed: 24-Apr-2018].
- [10] Toriello Nájera Álvaro, “Importancia de la Tecnología Computacional en las Organizaciones Empresariales para la Toma de Decisiones,” Atlantic international university school of business and economics, 2007.
- [11] R. S and Pressman, *Ingeniería del software un enfoque Práctico*, Séptima. Mexico, 2010.
- [12] A. Arias, *Aprende a programar ASP.NET y C#*. CreateSpace Independent Publishing Platform, 2015.
- [13] V. Gauchi Risso, “Estudio de los métodos de investigación y técnicas de recolección de datos utilizadas en bibliotecología y ciencia de la información,” *Rev. española Doc. Científica*, vol. 40, no. 2, p. 175, 2017.
- [14] G. C. Ng, “A Study of an Agile Methodology with Scrum Approach to the Filipino Company-Sponsored I.T. Capstone Program,” *Int. J. Comput. Sci. Res.*, vol. 2, no. 2, pp. 68–88.
- [15] T. Otzen and C. Manterola, “Técnicas de Muestreo sobre una Población a Estudio Sampling Techniques on a Population Study,” 2017.
- [16] R. Hernández Sampieri, C. Fernández Collado, and M. del P. Baptista Lucio, *metodología de la investigación*, McGRAW-HILL. 2014.
- [17] IEEE, “IEEE 29148-2011 - ISO/IEC/IEEE International Standard - Systems and software engineering -- Life cycle processes -- Requirements engineering,” *IEEE*, 2011. [Online]. Available: <https://standards.ieee.org/standard/29148-2011.html>. [Accessed: 18-Jul-2019].
- [18] P. Kruchten, “Planos Arquitectónicos: El Modelo de ‘4+1’ Vistas de la Arquitectura del Software *,” *IEEE Softw.*, vol. 12, 1995.
- [19] T. Dimes and M. Jimenez, *Conceptos Básicos De Scrum : Desarrollo De Software Agile Y Manejo De Proyectos Agile*. 2015.
- [20] A. N. Cadavid, J. D. Fernández Martínez, and J. Morales Vélez, “Revisión de metodologías ágiles para el desarrollo de software Revisión de metodologías ágiles para el desarrollo de software A review of agile methodologies for software development,” *Prospect*, vol. 11, pp. 30–39, 2013.
- [21] J. L. Aristegui, “Test cases in software test los casos de prueba en la prueba del software,” *Rev. Digit. Lámpsakos*, vol. 3, pp. 27–34, 2010.
- [22] G. A. García Mireles *et al.*, “Prácticas dirigidas a la mejora de la usabilidad del software políticas de ciencia y tecnología,” *Epistemos*, vol. 16, pp. 64–71, 2014.
- [23] ISO/IEC, “ISO/IEC 25010:2011(en), Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models,” *ISO/IEC*, 2011. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en>. [Accessed: 18-Jul-2019].
- [24] F. G. Ortiz Uribe, *Diccionario de metodología de la investigación científica*. Limusa, Noriega, 2003.