

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas
Carrera de Ingeniería en Sistemas Computacionales

ANÁLISIS DESCRIPTIVO DE LA IMPLEMENTACIÓN DE GENERADORES DE CÓDIGOS QR PARA DIFERENTES LENGUAJES DE PROGRAMACIÓN.

Trabajo de grado previo a la obtención del título de ingeniera
en Sistemas Computacionales

Autor:

Anderson Israel Vivas Padilla

Director:

Msc. Rea Peñafiel Xavier Mauricio

Ibarra – Ecuador

Julio 2019



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información.

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	100374414-9		
APELLIDOS Y NOMBRES:	VIVAS PADILLA ANDERSON ISAREL		
DIRECCIÓN:	IMBAYA (ANTONIO ANTE)		
EMAIL:	aivivasp@utn.edu.ec		
TELÉFONO FIJO:	(062) 540-081	TELÉFONO MÓVIL:	0993494321

DATOS DE LA OBRA	
TÍTULO:	ANÁLISIS DESCRIPTIVO DE LA IMPLEMENTACIÓN DE GENERADORES DE CÓDIGOS QR PARA DIFERENTES LENGUAJES DE PROGRAMACIÓN
AUTOR:	VIVAS PADILLA ANDERSON ISRAEL
FECHA:	29/07/2019
PROGRAMA:	PREGRADO
TÍTULO POR EL QUE OPTA:	INGENIERA EN SISTEMAS COMPUTACIONALES
DIRECTOR:	MSc. REA PEÑAFIEL XAVIER MAURICIO
ASESOR:	ING. VÍCTOR CARANQUI
ASESOR:	ING. FAUSTO SALAZAR

2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de esta y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 29 días del mes de Julio del 2019



.....
Nombre: Anderson Israel Vivas Padilla
Cédula: 100374414-9



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

Ibarra, 26 de febrero del 2019

CERTIFICACIÓN DIRECTOR DE TESIS

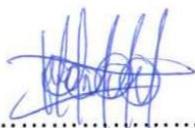
Por medio del presente yo MSc. Rea Peñafiel Xavier Mauricio, certifico que el trabajo de grado **“ANÁLISIS DESCRIPTIVO DE LA IMPLEMENTACIÓN DE GENERADORES DE CÓDIGOS QR PARA DIFERENTES LENGUAJES DE PROGRAMACIÓN”**, ha sido desarrollado en su totalidad por la Sñr Anderson Israel Vivas Padilla, portadora de cédula de identidad Nro. 1003744149.

Es todo cuento puedo certificar en honor a la verdad.

.....
MSc. MAURICIO REA
DIRECTOR DE TRABAJO DE GRADO

AUTORÍA

Yo, **ANDERSON ISRAEL VIVAS PADILLA**, portador de la cédula de ciudadanía número **1003744149**, declaro bajo juramento que el trabajo aquí descrito es de mi autoría, **ANÁLISIS DESCRIPTIVO DE LA IMPLEMENTACIÓN DE GENERADORES DE CÓDIGOS QR PARA DIFERENTES LENGUAJES DE PROGRAMACIÓN** que no ha sido previamente presentada para ningún grado, ni calificación profesional, y que se han respetado las diferentes fuentes y referencias.

A handwritten signature in blue ink, consisting of several vertical strokes and a horizontal line, positioned above a dotted line.

ANDERSON ISRAEL VIVAS PADILLA

C.I 1003744149

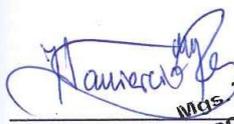
Ibarra, 22 de julio de 2019

**Ingeniero
Pedro Granda
COORDINADOR DE LA CARRERA CISIC/CSOFT**

Por medio de la presente me permito informar que el software producto de la elaboración de la tesis “Análisis descriptivo de la implementación de generadores de códigos QR para diferentes lenguajes de programación” realizado por el señor Anderson Israel Vivas Padilla con cédula de identidad número 1003744149, se encuentra instalado en el servidor de la carrera, ya que es de utilidad tanto como caso de estudio, así como de apoyo para temas de clase relacionados a aplicaciones web y uso de códigos QR.

El software se encuentra funcional y el código fuente se ha registrado en el repositorio de versionamiento de proyectos de software de la carrera.

Atentamente,


MgS. Mauricio Reyes
DOCENTE FICA UTM

Ing. Mauricio Reyes
DIRECTOR TESIS

AGRADECIMIENTO

Agradezco a mi madre por haberme dado la vida, y también por darme el apoyo para que yo pueda culminar mi carrera, es una de las personas más importantes que tengo. Además, agradecerle por todos los valores que me a infundido ya que han sido de mucha ayuda para que yo pueda culminar la carrera sin inconveniente.

A mi asesor de tesis el Ingeniero Mauricio Rea, por su amistad y orientación en la realización de mi tesis.

A mis amigos, Day, Cris, Tefy, Sebas, David y Mauricio, por brindarme su amistad en esta etapa de mi vida, y además agradecer los buenos momentos vividos, ya que han hecho que la universidad sea una bonita experiencia.

Anderson V.

TABLA DE CONTENIDOS

RESUMEN	XV
ABSTRACT	XVI
INTRODUCCIÓN	XVII
Tema	XVII
Problema	XVII
Antecedentes	XVII
Situación Actual	XVII
Prospectiva	XIX
Planteamiento del problema.	XIX
Objetivos	XX
Objetivo General	XX
Objetivos Específicos.....	XX
Alcance.....	XXI
Justificación	XXI
Contexto	XXII
CAPÍTULO 1	21
1.1 Análisis Descriptivo.....	21
1.2 Implementación	21
1.3 Códigos QR	22
1.3.1 Origen de los Códigos QR.....	22
1.3.2 Aplicaciones del código QR.....	23
1.3.3 Como generar un código QR.....	24
1.4 Lenguajes de Programación.....	24
1.4.1 Lenguaje de programación Java	24
1.4.2 Lenguaje de programación C#	26

1.4.3	Lenguaje de programación JavaScript	27
1.4.4	Lenguaje de Programación PHP	29
1.5	Librerías de programación para generar códigos QR	30
1.5.1	¿Qué es una librería de programación?	30
1.5.2	Librería de programación para el lenguaje Java	31
1.5.3	Librería de programación para el lenguaje C#	31
1.5.4	Librería de programación para el lenguaje JavaScript	32
1.5.5	Librería de programación para el lenguaje PHP	32
1.6	ISO 25010.....	32
CAPÍTULO 2		35
2.1	Metodología de desarrollo	35
2.1.1	Modelo en cascada	35
2.2	Requisitos	35
2.3	Diseño.....	36
2.4	Implementación de componentes de Software.....	36
2.4.1	Implementación en el lenguaje Java	36
2.4.2	Implementación en el lenguaje C#	42
2.4.3	Implementación en el lenguaje Java Script	47
2.4.4	Implementación en el lenguaje PHP	53
2.5	Comparativa en base a la característica 2 de la ISO 25010	57
2.5.1	Comportamiento de tiempo	57
2.5.2	Utilización de recursos	58
2.5.3	Capacidad	60
2.6	Análisis de implementación	61
CAPÍTULO 3		62
3.1	Pruebas	62
3.1.1	Comportamiento de tiempo	62
3.1.2	Utilización de recursos.	64

3.1.3	Capacidad.....	66
3.2	Análisis de Resultados	68
3.3	Tabla resumen del análisis descriptivo.....	71
3.4	Análisis de impacto.....	71
	CONCLUSIONES	73
	RECOMENDACIONES.....	74
	BIBLIOGRAFÍA Y LINKOGRAFÍA.....	75

ÍNDICE DE FIGURAS

Figura. 1 Árbol de Problema	XX
Figura. 2 Ejemplo de un Código QR	22
Figura. 3 Logo de Java	24
Figura. 4 Logo de C#	26
Figura. 5 Logo de JavaScript	27
Figura. 6 Logo de PHP	29
Figura. 7 Características de la ISO/IEC 25010.....	33
Figura. 8 Diseño del Modelo en Cascada.....	35
Figura. 9 Creación de un proyecto EAR Java (JSF)	37
Figura. 10 Módulos en aplicaciones EAR Java (JSF).....	38
Figura. 11 Estructura del Módulo Model en Java (JSF).....	38
Figura. 12 Estructura del Módulo Controller en Java (JSF)	40
Figura. 13 Estructura del Módulo View en Java (JSF).....	41
Figura. 14 Proceso de generación de un código QR con la librería Primefaces 7.0	42
Figura. 15 Proyecto creado en C#	43
Figura. 16 Estructura del Módulo Models en C#(.NET)	44
Figura. 17 Estructura del Módulo Controller en C# (.NET)	45
Figura. 18 Estructura del Módulo View en C# (.NET)	46
Figura. 19 Proceso de generación de un código QR con la librería MessagingToolkitQRCode.....	47
Figura. 20 Creación de carpeta Express Nodejs	48
Figura. 21 Estructura del Módulo Model con JavaScript (Express)	49
Figura. 22 Estructura del Módulo Controller con JavaScript (Angular)	50
Figura. 23 Estructura del Módulo View con JavaScript (Angular).....	52
Figura. 24 Proceso de generación de un código QR con la librería angularx-qrcode ..	53

Figura. 25 Carpetas creadas en el proyecto PHP.....	54
Figura. 26 Estructura del Módulo Model en php.	54
Figura. 27 Estructura del Módulo Controller en php.	55
Figura. 28 Estructura del Módulo View en php.	56
Figura. 29 Proceso de generación de un código QR con la librería PhpqrCode.	57
Figura. 30 Tiempos de Espera en los diferentes Lenguajes de Programación.	63
Figura. 31 Tiempos de Respuesta en los diferentes Lenguajes de Programación.....	63
Figura. 32 Datos de Rendimiento en los diferentes Lenguajes de Programación.	64
Figura. 33 Utilización de CPU en los diferentes lenguajes de Programación.....	65
Figura. 34 Utilización de memoria en los diferentes Lenguajes de Programación	65
Figura. 35 Número de peticiones online en los diferente Lenguajes de Programación	66
Figura. 36 Número de accesos simultáneos en los diferentes Lenguajes de Programación	67
Figura. 37 Sistema de transmisión de ancho de banda en los diferentes Lenguajes de Programación.	67
Figura. 38 Evaluación de eficiencia de desempeño con el lenguaje Java.....	68
Figura. 39 Evaluación de eficiencia de desempeño con el lenguaje C#.....	69
Figura. 40 Evaluación de eficiencia de desempeño con el lenguaje JavaScript.....	69
Figura. 41 Evaluación de eficiencia de desempeño con el lenguaje PHP.....	70
Figura. 42 Comparativa de Resultados entre lenguajes de programación.	70

ÍNDICE DE TABLAS

TABLA 1 Características del Lenguaje de programación Java.....	25
TABLA 2 Características del Lenguaje de Programación C#	27
TABLA 3 Características del Lenguaje de Programación JavaScript	29
TABLA 4 Características del Lenguaje de Programación PHP	30
TABLA 5 Sub-características de eficiencia de desempeño	34
TABLA 6 Descripción de los elementos del módulo Model en Java (JSF)	39
TABLA 7 Descripción de los elementos del módulo Controller en Java (JSF)	40
TABLA 8 Descripción de los elementos del módulo View en Java (JSF)	41
TABLA 9 Descripción de los elementos del módulo Models en C# (.NET)	44
TABLA 10 Descripción de los elementos del módulo Controller en C# (.NET)	45
TABLA 11 Descripción de los elementos del módulo View en C# (.NET)	46
TABLA 12 Complementos de programación JavaScript (Express)	48
TABLA 13 Descripción de los elementos del módulo Model en JavaScript (Express) .	49
TABLA 14 Descripción de los elementos del módulo Controller en JavaScript (Angular)	50
TABLA 15 Descripción de los elementos del módulo View en JavaScript.....	52
TABLA 16 Descripción de los elementos del Módulo Model en php	54
TABLA 17 Descripción de los elementos del Módulo Controller en php.....	55
TABLA 18 Descripción de los elementos del Módulo View en php	56
TABLA 19 Ventajas y desventajas de implementación.....	61
TABLA 20 Tabla resumen del análisis descriptivo.....	71

RESUMEN

En la actualidad los códigos QR en el Ecuador son utilizados como una herramienta de marketing, y se ve una escasa utilización de estos en aplicaciones web, y no lo ven como una opción para automatización de procesos.

Con el avance tecnológico que se da en la actualidad las organizaciones han optado por la automatización de procesos, sustituyendo los procesos manuales con aplicaciones de software, con el objetivo principal de reducir los costos mediante la integración de aplicaciones, reduciendo la mano de obra y acelerando el tiempo de ejecución de las actividades. La tecnología del código QR es una herramienta que se ha utilizado como una alternativa para la automatización de procesos en diferentes áreas como Educación, Industria, Medicina y Turismo.

En este trabajo se realiza un análisis descriptivo de generadores de códigos QR en base a la característica 2 de la ISO 25010 que trata sobre eficiencia y desempeño del software, de manera que el estudio sirva de base para futuros desarrolladores de software utilizando PHP, Java, JavaScript y C#. La característica de la ISO mencionada representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones.

Para esto se analiza los complementos (librerías) que existen para generar códigos QR en los lenguajes de programación PHP, Java, JavaScript y C#, y se realiza la implementación de estos componentes y librerías de software en base a la arquitectura MVC.

Se valida los resultados y se obtiene una tabla resumen del análisis descriptivo que sirva como guía rápida para los desarrolladores de software.

ABSTRACT

Currently QR codes in Ecuador are used as a marketing tool, and you see little use of these in web applications, and it's not seen as an option for process automation.

With the technological advancement that is currently being made, organizations have opted for process automation, replacing manual processes with software applications, with the main goal of reducing costs by integrating applications, reducing labor and accelerating the execution time of activities. QR code technology is a tool that has been used as an alternative for process automation in different areas such as Education, Industry, Medicine and Tourism.

In this work a descriptive analysis of QR code generators is carried out based on ISO 25010 characteristic 2 about software efficiency and performance, so that the study will serve as a basis for future software developers using PHP, Java, JavaScript y C#. The ISO characteristic mentioned represents the performance relative to the amount of resources used under certain conditions.

For this we analyze the plugins (libraries) that exist to generate QR codes in the programming languages PHP, Java, JavaScript y C#, and the implementation of these components and software libraries is carried out based on the MVC architecture.

The results are validated, and you get a summary table of descriptive analysis that serves as a quick guide for software developers.

INTRODUCCIÓN

Tema

Análisis descriptivo de la implementación de generadores de códigos QR para diferentes lenguajes de programación.

Problema

Antecedentes

Con el avance tecnológico que se da en la actualidad las organizaciones han optado por la automatización de procesos, sustituyendo los procesos manuales con aplicaciones de software, con el objetivo principal de reducir los costos mediante la integración de aplicaciones, reduciendo la mano de obra y acelerando el tiempo de ejecución de las actividades.(Heflo, 2017)

La tecnología del código QR es una herramienta que se ha utilizado como una alternativa para la automatización de procesos en diferentes áreas como Educación, Industria, Medicina y Turismo. Los Códigos QR fueron presentados en 1994 por Denso Wave, subsidiaria japonesa en el grupo Toyota, después de medio siglo de uso de los códigos de barras se da la aparición de estos códigos que almacenan más información y por lo tanto le encuentran más aplicaciones en la actualidad.(Huidrobo, 2009)(UnitagQR, 2011)

Los códigos QR permiten interactuar con el mundo a través de su smartphone y han estado en constaten evolución, hay diferentes tipos de QR que son: Aztec code, Micro QR code, Datamatrix, Microsoft Tag.(UnitagQR, 2011).Para la generación de estos códigos en aplicaciones web existen complementos (librerías) para los principales lenguajes de programación. Los lenguajes que ayudarán a realizar el análisis descriptivo serán PHP, Java, JavaScript, C#. Esta lista se ha obtenido en base a anuncios en webs para encontrar trabajo, así como también en GitHub, ya que GitHub entrega estadísticas de los lenguajes de programación más utilizados en la actualidad.

Situación Actual

Actualmente son varios los países que elevan su producción gracias a las TIC. En Ecuador esta apropiación se está realizando en varias etapas que estimulan la micro, mediana y grande empresa. (MINTEL, 2017)

En este marco, el MINTEL se enfoca en promover políticas públicas para involucrar en las TIC a los diferentes sectores de la sociedad, entre los que se incluyen alcaldías, universidades y sector empresarial.(MINTEL, 2017)

En el Ecuador la tecnología del código QR es utilizada para operaciones de marketing, estos se pueden encontrar con información de publicidad en folletos, carteles, revistas, etc. No obstante, esta tecnología no se la utiliza en la automatización de procesos por parte de los diferentes sectores de la sociedad del Ecuador.

Los códigos QR en otros países son de gran ayuda en la automatización de procesos en diferentes sectores, un ejemplo de esto es Tailandia: La sobrepoblación de los hospitales ha sido un problema en el sistema público de salud tailandés.(Charoensiriwath, Surasvadi, Pongnumkul, & Pholprasit, 2015) Para aliviar el problema se utiliza el código QR y una aplicación de teléfono móvil para capturar los procesos hospitalarios y el tiempo dedicado a ocupaciones con los datos de disponibilidad del personal del hospital. (Charoensiriwath et al., 2015) La aplicación permite al personal del hospital para registrar varias actividades por sí mismos. Un objetivo de este sistema es permitir que las actividades se registren en cualquier lugar sin necesidad de una computadora. (Charoensiriwath et al., 2015)

Los códigos QR en el área del turismo son de gran utilidad, ya que ayudan a optimizar los servicios en los establecimientos turísticos como museos, en la Pontificia Universidad Católica de Chile se ha hecho un análisis para ver el impacto que tiene los códigos QR versus pantallas tradicionales para proporcionar información sobre las exhibiciones en un Museo.(Perez-Sanagustín, Parra, Verdugo, Gonzalo García-Galleguillos, & Nussbaum, 2016) Los códigos QR bidireccionales son una alternativa económica para mostrar contenido digital en museos, especialmente para los visitantes en edad universitaria.(Perez-Sanagustín et al., 2016).

Estudiantes de la Universidad Técnica del Norte, de la Carrera de Ingeniería en Sistemas Aplicada, presentaron el proyecto QRegistroRapido, el cual tiene como objetivo principal establecer una técnica alternativa para el registro de los asistentes a un evento mediante el uso de códigos QR. Resolviendo la problemática que se presenta en la mayoría de los eventos que implican la asistencia de personas, se presentan inconvenientes que no permiten un proceso óptimo para el registro de asistencia. Al ser un proceso manual donde los asistentes se registran firmando y llenando hojas impresas con listados, conlleva a una serie de imprecisiones y lentitud.

Prospectiva

Con lo expuesto anteriormente se plantea realizar un análisis descriptivo de generadores de códigos QR en base a la característica 2 de la ISO 25010 que trata sobre eficiencia y desempeño del software, de manera que el estudio sirva de base para futuros desarrolladores de software utilizando Java, C#, JavaScript y PHP. La característica de la ISO mencionada representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. A su vez esta característica se divide en las siguientes subcaracterísticas:

- Comportamiento temporal. Los tiempos de respuesta y procesamiento y las ratios de throughput de un sistema cuando lleva a cabo sus funciones bajo condiciones determinadas en relación con un banco de pruebas (benchmark) establecido.
- Utilización de recursos. Las cantidades y tipos de recursos utilizados cuando el software lleva a cabo su función bajo condiciones determinadas.
- Capacidad. Grado en que los límites máximos de un parámetro de un producto o sistema software cumplen con los requisitos.

Con este análisis se aportará al cumplimiento del objetivo 9 en el inciso 9.5 de la agenda 2030 y los Objetivos de Desarrollo Sostenible una oportunidad para América Latina y el Caribe publicado por la UNESCO y ONU en mayo de 2016, objetivo el cual se enfoca en la Industria, Innovación e Infraestructura.(Unidas & CEPAL, 2016) De la misma forma va de acuerdo con el Objetivo No.5 del Plan Nacional de Desarrollo Toda una vida 2017-2021 publicado por la Secretaria Nacional de Planificación y Desarrollo Ecuador, el cual hace énfasis en Impulsar la productividad y competitividad para el crecimiento Económico sostenible de manera de manera retributiva y solidaria.

Con este análisis también se pretende dejar una tabla resumen del análisis descriptivo sobre los complementos (librerías) de códigos QR en los diferentes lenguajes de programación, la cual ayudara a los desarrolladores de software a elegir la mejor opción para realizar aplicaciones web que generen códigos QR con información específica.

Este análisis además promoverá la innovación en todos los sectores de la sociedad del Ecuador ya que presenta una nueva opción para automatización de procesos.

Planteamiento del problema.

En la actualidad los códigos QR en el Ecuador son utilizados como una herramienta de marketing, y se ve una escasa utilización de estos en aplicaciones web, y no lo ven como una opción para automatización de procesos.

Tecnológico

Al momento no existe un análisis descriptivo el cual muestre resultados estadísticos mediante una ISO, sobre cuál es el mejor lenguaje de programación y complemento (librería) para generar códigos QR en aplicaciones web.

Social

Ausencia de utilización de la tecnología del código QR en aplicaciones Web

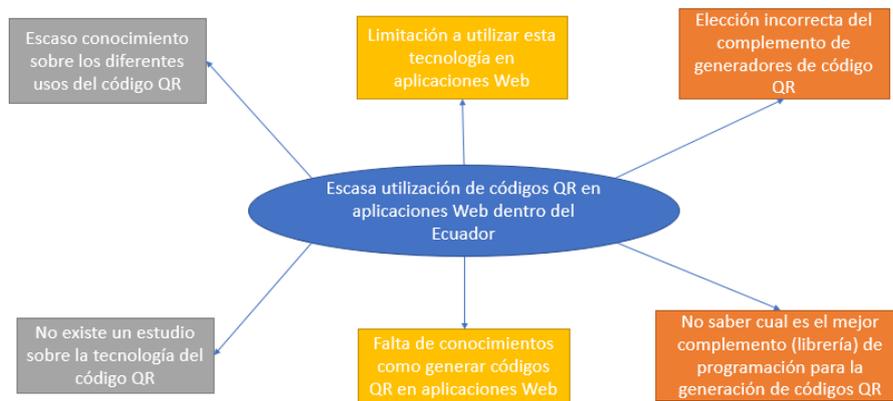


Figura. 1 Árbol de Problema
Fuente: Propia

Objetivos

Objetivo General

Realizar un análisis descriptivo de la implementación de generadores de códigos QR para diferentes lenguajes de programación.

Objetivos Específicos

- Analizar los complementos (librerías) que existen para generar códigos QR en los lenguajes de programación Java, C#, JavaScript y PHP, mediante la ISO 25010.
- Implementar componentes y librerías de software en base a la arquitectura MVC en los lenguajes de programación Java, C#, JavaScript y PHP.
- Obtener una tabla resumen del análisis descriptivo que sirva como guía rápida para los desarrolladores de software.
- Validar los resultados.

Alcance

Los resultados de este análisis descriptivo estarán dirigidos a toda la comunidad de desarrolladores de software, ya que se tendrán componentes reutilizables y datos estadísticos de las mejores opciones para generar códigos QR en aplicaciones web, el análisis se realizará con diferentes lenguajes de programación.

Implementar componentes de software en base a la arquitectura MVC para generar códigos QR con los lenguajes de programación: Java, JavaScript, PHP y C# (.Net), mediante esta implementación se obtendrá demos los cuales servirán para obtener datos comparativos en base a la característica 2 de la ISO 25010, se obtendrá los siguientes resultados:

- Comportamiento temporal al momento de generar el código QR
- Utilización de recursos al momento de generar el código QR.
- Capacidad del lenguaje de programación para la generación de códigos QR.

Luego de la implementación de los componentes de software se determinará el método estadístico adecuado que permitirá validar los resultados de los demos obtenidos, para la validación de resultados se utilizará la información de la base de datos de los estudiantes de la carrera de ingeniería en sistemas computacionales.

Justificación

Social

El presente proyecto va a tener un enfoque hacia los objetivos de desarrollo sostenible planteado por la ONU y UNESCO: INDUSTRIA, INNOVACIÓN E INFRAESTRUCTURA (Unidas & CEPAL, 2016) literal 9.5 Aumentar la investigación científica y mejorar la capacidad tecnológica de los sectores industriales de todos los países, en particular los países en desarrollo, entre otras cosas fomentando la innovación y aumentando considerablemente, de aquí a 2030, el número de personas que trabajan en investigación y desarrollo por millón de habitantes y los gastos de los sectores público y privado en investigación y desarrollo.(Unidas & CEPAL, 2016)

De la misma manera se relaciona con el PLAN NACIONAL DE DESARROLLO TODA UNA VIDA(SEMPLADES, 2017) (Enríquez, 2016)promulgado en la Constitución de la República del Ecuador, enfocándose principalmente en Promover la investigación, la formación, la capacitación, el desarrollo y la transferencia tecnológica, la innovación y el

emprendimiento, la protección de la propiedad intelectual, para impulsar el cambio de la matriz productiva mediante la vinculación entre el sector público, productivo y las universidades.(SEMPLADES, 2017)

Tecnológico

El código QR es una tecnología muy versátil la cual se le puede dar muchos usos, el avance tecnológico ha hecho que sea utilizada como una opción al momento automatizar algún proceso, en el Ecuador existe esta tecnología, pero no se la utiliza todavía automatizando procesos en los diferentes sectores sociales.

Contexto

En el portal BibliotecasdelEcuador.com que recoge y almacena los trabajos de investigación que se desarrollan en las distintas instituciones académicas ecuatorianas no existen trabajos que contengan información sobre la automatización de procesos con la tecnología del código QR.

CAPÍTULO 1

Marco Teórico

1.1 Análisis Descriptivo

Un análisis descriptivo se enfoca en analizar los datos relevantes sobre un área o un tema específico de investigación, con el fin de encontrar patrones, diferencias significativas con referencia a temas similares y relación entre variables del mismo tema de investigación o con otros, en base a ello, se organiza, tabula y describe el resultado (QuestionPro, 2017).

Para obtener un análisis descriptivo se debe hacer cálculos de medidas simples de composición y distribución de variables. Dependiendo del tipo de datos, pueden ser proporciones, tasas, razones o promedios (QuestionPro, 2017). Los resultados de este tipo de investigación pueden ser utilizado para hacer un análisis cuantitativo.

En este trabajo se investigará generadores de códigos QR y se implementará en los diferentes lenguajes de programación, posteriormente se obtendrá los datos resultantes de hacer las pruebas, esto en base a la ISO 25010. Así, se concretará el Análisis Descriptivo aplicado a este tema de investigación.

1.2 Implementación

Una implementación es la ejecución de un proceso, ya sea, de una aplicación informática, un plan, estándar, algoritmo o política(Wikipedia, 2019).

En el área de Ingeniería de software una implementación es la realización de un algoritmo como un programa, y la integración de componentes de software, en base a especificaciones y requisitos del software(Wikipedia, 2019).

En este trabajo se implementará diferentes componentes de software desarrollados para la generación de código QR, con las herramientas, requisitos y arquitectura que requiera cada lenguaje de programación propuesto.

1.3 Códigos QR

1.3.1 Origen de los Códigos QR.

El código de barras está presente en la mayoría de los artículos que adquirimos, estos almacenan un código numérico único, con el cual se puede identificar a cada producto. Por ejemplo, para el proceso de cobro en un supermercado, los cajeros pasan cada producto por un lector de códigos de barras, así se obtiene el precio y el cliente lo puede ver. Esta tecnología facilita el cobro de artículos evitando que haya errores humanos, que además pueden producirse en muchos otros procesos (Huidrobo, 2009).

En 1994 aparecen los códigos QR, los cuales tienen mayor capacidad de almacenamiento y son más fáciles de utilizar, ya que estas tecnologías pueden ser interpretables por cualquier dispositivo que pueda captar imágenes y cuenten con el software adecuado. Fueron creados por la filial de Toyota Denso-Wave, y es muy eficiente para codificar caracteres. Pueden almacenar hasta 7.089 caracteres y sus posibilidades son innumerables, desde codificar información sobre webs, promociones, publicidad, organización de inventarios, mensajes cortos, entre otros (Huidrobo, 2009). Razón por la cual, han sido utilizados en muchas aplicaciones apropiadas para tratar con datos.(Zhou, Hu, Yuan, Zhang, & Huo, 2018)

Visualmente un código QR se conforma de cuadros blancos y negros, dichos cuadros son parte de su estructura, el tamaño de estos varía en proporción a la cantidad de información que se almacene, es decir que, mientras más información se almacene, el número de cuadros aumenta y su tamaño se reduce (Zhou et al., 2018).



Figura. 2 Ejemplo de un Código QR
Fuente: Propia

1.3.2 Aplicaciones del código QR

Con los avances tecnológicos y la popularización de los teléfonos inteligentes, el código QR cada vez tiene más aplicaciones en la fabricación, ventas, logística, almacenaje, transporte e industria de productos de consumo (Zhou et al., 2018). De la misma forma se le está dando más uso al momento de automatizar procesos en áreas como la Educación, Industria, Medicina, Innovación y Turismo.

El aporte que han tenido los códigos QR se ha hecho presente en la educación y ha sido principalmente en el proceso de enseñanza-aprendizaje como una herramienta pedagógica. Un claro ejemplo se presenta en el artículo *Compressed QR code- based mobile voice guidance service for the visually disabled* el cual demuestra la usabilidad de esta tecnología al generar información comprimida para varios contenidos de lectura (libros, revistas, afiches) (Kim et al., 2018).

Cuando se menciona áreas como la Industria se hace referencia a toda actividad que tiene como objetivo la transformación de materia prima en productos, en esta área los códigos QR han sido de mucha importancia puesto que han ayudado a mejorar y controlar algunos procesos. Un claro ejemplo de esto es la empresa Automation Colombia, que se dedica principalmente a la venta de equipos industriales y a la automatización de procesos, ha implementado esta tecnología para la identificación de los equipos que suministra con sus características específicas (Pineda Bohórquez, 2014).

En áreas como la medicina este código ha sido de gran utilidad, ya que se ha implementado esta tecnología con el objetivo de agilizar y optimizar procesos. Una aplicación de esto es en el sistema público tailandés, la sobrepoblación en los hospitales de este país es un problema por lo que las autoridades pertinentes sobre el tema han optado por utilizar los códigos QR y una aplicación de teléfono móvil como una solución, para capturar los procesos hospitalarios y el tiempo dedicado a ocupaciones con los datos de disponibilidad del personal del hospital. Uno de los objetivos de este sistema es permitir que las actividades se registren automáticamente después de leer el código QR, con esto ahorrar tiempo y abastecer la aglomeración de personas (Charoensiriwath et al., 2015).

1.3.3 Como generar un código QR

En la actualidad existen aplicaciones web y de escritorio las cuales ayudan a generar estos códigos, pero en su mayoría se limitan solo a generar URLs, una de las webs en donde se puede realizar la codificación es <https://es.qr-code-generator.com/> .

A medida que avanza la tecnología, aumenta la utilización de códigos QR dentro de las aplicaciones Web, para el uso de esta tecnología existen componentes o librerías que facilitan la implementación de estos con diferentes herramientas y en diferentes lenguajes de programación. Sin embargo, hasta el momento no existe una investigación que establezca cual es el mejor lenguaje de programación y la mejor librería para generar códigos QR.

Por tal motivo este trabajo tiene como objetivo principal, realizar un análisis descriptivo de la implementación de generadores de códigos QR para diferentes lenguajes de programación.

1.4 Lenguajes de Programación

Los lenguajes de programación son conocidos como idiomas artificiales, los cuales están diseñados para expresar una serie de instrucciones que son interpretadas y ejecutadas por ordenadores. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, expresar algoritmos con precisión y además, es una forma de comunicación humana (EcuRed, s/f).

Un lenguaje de programación está formado de un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila y se mantiene el código fuente de un programa informático se le llama programación (EcuRed, s/f).

1.4.1 Lenguaje de programación Java

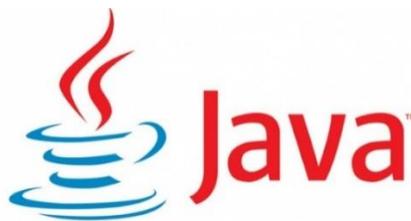


Figura. 3 Logo de Java

Fuente: <https://mundocontact.com/oracle-gana-litigio-google-derechos-java/java-logo/>

Este lenguaje fue muy importante para la evolución de la programación. La primera versión que salió es java 1.0, fue creada por Sun Microsystems en el año de 1995. Este hecho causó una revolución, ya que transformó de manera radical la Web y lo convirtió en un entorno altamente interactivo. Con el paso del tiempo, este lenguaje se estableció como un estándar en el diseño de lenguajes de computadoras (Schildt, 2007). Con los años, Java siguió creciendo, evolucionando y redefiniéndose en distintas formas, a diferencia de muchos otros lenguajes que se muestran lentos para incorporar nuevas características. La primera actualización importante de Java fue la versión 1.1, esta versión añadía y eliminaba atributos, redefiniendo la manera en que se manejaban los eventos y reconfigurando muchas características de la biblioteca original (Schildt, 2009).

La siguiente generación fue java 2.0, esta versión marcó el inicio de la “era moderna”, con esto, Sun Microsystems empaquetó su producto como J2SE (Java 2 Platform Standard Edition) y aplicó el número de versión. La segunda revolución de esta esta tecnología fue J2SE 5, la cual tenía muchas funciones nuevas que cambiaron de manera fundamental el carácter del lenguaje, aumentando su capacidad y su alcance, para tener una idea de los alcances obtenidos con los nuevos cambios originados por J2SE 5, aquí una lista de las adiciones importantes realizadas (Schildt, 2007):

- Elementos genéricos
- Auto encuadre/des encuadre
- Enumeraciones
- El bucle mejorado for del estilo “for-each”
- Argumentos de longitud variable (varargs)
- Importación estática
- Metadatos (anotaciones)

TABLA 1 Características del Lenguaje de programación Java

Lenguaje de programación Java	
Característica	Detalle
Simple	Java tiene un conjunto conciso y cohesivo de funciones que facilitan su aprendizaje y uso.(Schildt, 2007)
Seguro	Java proporciona un medio seguro de crear aplicaciones de Internet.(Schildt, 2007)
Portable	Los programas de Java pueden ejecutarse en cualquier entorno para el cual haya un sistema de Java en tiempo de ejecución.(Schildt, 2007)

Robusto	Los programas de Java pueden ejecutarse en cualquier entorno para el cual haya un sistema de Java en tiempo de ejecución.(Schildt, 2007)
Interpretado	Los programas de Java pueden ejecutarse en cualquier entorno para el cual haya un sistema de Java en tiempo de ejecución.(Schildt, 2007)
Dinámico	Los programas de Java pueden ejecutarse en cualquier entorno para el cual haya un sistema de Java en tiempo de ejecución.(Schildt, 2007)

Fuente: Propia

1.4.2 Lenguaje de programación C#



Figura. 4 Logo de C#

Fuente: <https://github.com/exercism/meta/issues/27>

El lenguaje C# es descendiente directo de C, C++ y Java. C marcó el inicio de la era moderna en la programación, fue creado por Dennies Ritchie en la década de 1970. En esta época ya existía Pascal que además tenía mucho éxito, hasta que apareció C, que al ser un lenguaje estructurado tuvo más éxito, debido a que cuenta con un rico conjunto de declaraciones de control bien diseñadas, bloques de código y otras mejoras que facilitan la organización y el mantenimiento del programa (McGraw-Hill, 2010).

A finales del año 1970 los programas se volvían demasiados grandes, debido a que el tamaño de muchos proyectos estaba en los límites de lo que las metodologías de la programación estructurada y el lenguaje C podían manejar. Para resolver este problema se crea una nueva forma de programar denominada “programación orientada a objetos”, en base a esto Bjarne Stroustrup creo C++, agrego nuevas características al lenguaje de programación C con el fin de que tenga la capacidad de soportar la nueva metodología (McGraw-Hill, 2010).

Con el surgimiento de Internet, Java carecía de una característica muy importante que era la integración con la plataforma Windows, es decir, no se podía programar en esta plataforma.

Sin embargo, los programas creados en este lenguaje pueden ejecutarse en el ambiente Windows (siempre y cuando se encuentre instalada la Máquina Virtual Java). Con estos antecedentes y otras necesidades Microsoft decide en 1990 desarrollar C#, como parte de su estrategia global .NET. La primera versión alfa fue lanzada a mediados de 2000 y el arquitecto encargado de su desarrollo fue Anders Hejlsberg (McGraw-Hill, 2010).

Desde su lanzamiento original, C# ha evolucionado rápidamente. Poco después del primer lanzamiento, Microsoft lanzó la versión 1.1, a la cual no se le añadieron mayores características y los ajustes fueron mínimos. Actualmente se encuentra en la versión 7.0, esta cuenta con funciones y características mejores que las de versiones anteriores.

TABLA 2 Características del Lenguaje de Programación C#

Lenguaje de Programación C#	
Característica	Detalle
Sencillez de uso	C# elimina muchos elementos añadidos por otros lenguajes y facilitan su uso y comprensión.(Violeta_1879354, 2008)
Orientado a objetos	C# como lenguaje de última generación, y de propósito general, es orientado a objetos.(Violeta_1879354, 2008)
Orientado a componentes	La propia sintaxis de C# incluye elementos propios del diseño de componentes que otros lenguajes tienen que simular.(Violeta_1879354, 2008)
Instrucciones seguras	Para evitar errores comunes como se producían programando en otros lenguajes, en C# se han impuesto una serie de restricciones en el uso de instrucciones de control más comunes.(Violeta_1879354, 2008)
Compatible	Para facilitar la migración de programadores de C++ o Java a C#, no sólo se mantiene una sintaxis muy similar a la de los dos anteriores lenguajes.(Violeta_1879354, 2008)

Fuente: Propia

1.4.3 Lenguaje de programación JavaScript



Figura. 5 Logo de JavaScript

Fuente: <https://www.genbeta.com/desarrollo/javascript-funciones-y-expresiones>

Al principio de los años 90, la manera en que la mayoría de los usuarios se conectaba a internet era mediante la utilización de módems a una velocidad máxima de 28 kbps, en esta época aparecieron las aplicaciones web con formularios, las que cada vez eran más complejas. Surgió la necesidad de crear un lenguaje de programación que pueda ejecutarse en el navegador del usuario, lo que ayudaría a conocer al instante los posibles errores cuando se digita información, esto evitaría tener que esperar más tiempo hasta que el servidor muestre formulario indicando los errores producidos (Web, 2015).

En 1995 Brendan Eich, un programador que trabajaba en Netscape, lanzo el lenguaje de programación LiveScript, este se adaptaba a tecnologías de la época como ScriptEase. Eich pensó que este lenguaje podría solucionar los problemas que surgían en los formularios de las aplicaciones web. La primera versión fue lanzada con el navegador Netscape Navigator 2.0.(Web, 2015)

Posteriormente, Netscape firmó una alianza con Sun Microsystems para el desarrollo del nuevo lenguaje de programación. Antes del lanzamiento Netscape decidió cambiar el nombre por el de JavaScript. La razón del cambio de nombre fue exclusivamente por marketing.(Web, 2015)

Desde el principio, este lenguaje tuvo mucho éxito, por lo que cuando se lanzó Netscape Navigator 3.0 ya se incluía la versión 1.1. En esta época Microsoft lanzo el lenguaje JScript una copia del JavaScript. Lo que llevo a que Netscapet estandarizará su lenguaje con el estándar ECMA-262 de la organización internacional para estandarización (ISO).(UTEL, 2017)

JavaScript agrega mayor interactividad a la web, ya que está disponible principalmente en la parte visual (Front End), además se puede utilizar librerías y framework¹ escritas sobre JavaScript.(Grados, 2017)

¹ Framework: Es un entorno de trabajo, en el desarrollo de software es una estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software.

TABLA 3 Características del Lenguaje de Programación JavaScript

Lenguaje de Programación JavaScript	
Característica	Detalle
Multiplataforma	Ya que se puede utilizar en Windows, Linux o Mac o en el navegador de tu preferencia.(Grados, 2017)
Imperativo y estructurado	Mediante un conjunto de instrucciones indica al computador qué tarea debe realizar.(Grados, 2017)
Prototipado	Debido a que usa prototipos en vez de clases para el uso de herencia.(Grados, 2017)
Es Interpretado	No se compila ² para poder ejecutarse.(Grados, 2017)

Fuente: Propia

1.4.4 Lenguaje de Programación PHP



Figura. 6 Logo de PHP

Fuente: <https://es.wikipedia.org/wiki/PHP>

Este lenguaje de programación se viene elaborando desde el año 1994, en esta línea de tiempo hasta la actualidad, ha tenido que pasar por muchos cambios, el primer lanzamiento fue PHP/FI, la que carecía de muchas características por lo que se le consideró primitiva. En 1997 se hicieron mejoras para PHP, así, salió la versión 3.0, más potente y consistente, además contaba con arquitectura extensible que animaba a los desarrolladores independientes a crear sus propias versiones mejoradas y extensiones del lenguaje. En el año 2000 esta plataforma tuvo un cambio radical ya que ofrecía un nuevo motor, mejor rendimiento, más confiabilidad y soporte integrado para sesiones y características orientadas a objetos (Vaswani, 2010) .

² Compilar: traducir un código de programación a código ejecutable por la máquina.

En la actualidad PHP es muy utilizado en la WEB, se estima 100 millones de páginas desarrolladas con este lenguaje. A pesar del tiempo, sigue fiel a su nombre original: Personal Home Page, ya que proporciona el modo más sencillo para que sus páginas web cobren vida en un servidor. Este lenguaje se ha convertido en unos de los más profesionales, lo que lo hace adecuado para sitios de alto nivel (Holzner, 2009).

TABLA 4 Características del Lenguaje de Programación PHP

Lenguaje de Programación PHP	
Característica	Detalle
Rendimiento	El motor de PHP fue completamente rediseñado con un manejo óptimo de memoria para mejorar su rendimiento. Además, están disponibles aceleradores de terceros que pueden mejorar aún más el rendimiento.(Vaswani, 2010)
Portabilidad	PHP estaba disponible para UNIX, Microsoft Windows, Mac OS y OS/2 y el software desarrollado en PHP se pueden transportar de una plataforma a otra. Como resultado las aplicaciones PHP desarrolladas en Windows se podrán ejecutar en UNIX sin grandes contratiempos.(Vaswani, 2010)
Código Libre	Una de las mejores características que PHP es un proyecto de código libre, el lenguaje es desarrollado por un grupo de programadores distribuidos por todo el mundo, quienes ponen a disposición gratuita el código fuente a través de internet. Con ello se reduce el costo de desarrollo de software con este lenguaje sin afectar la flexibilidad ni la confiabilidad de los productos.(Vaswani, 2010)
Soporte a aplicaciones de terceros	PHP tiene soporte a una gama amplia de diferentes bases de datos entre las cuales se incluyen MySql, PostgresSQL y Oracle. Este lenguaje soporta más de 15 diferentes motores de base de datos, e incluye una API ³ común para el acceso de base de datos

Fuente: Propia

1.5 Librerías de programación para generar códigos QR

1.5.1 ¿Qué es una librería de programación?

Es un conjunto de subprogramas codificados en un lenguaje de programación específico, estas son usadas para el desarrollo de software. Una librería puede vincularse a un programa en distintos puntos del desarrollo o ejecución, según el tipo de vínculo que se quiera establecer.

³ API (Interfaz de programación de aplicaciones) Una API es un conjunto de funciones y procedimientos que cumplen una o muchas funciones con el fin de ser utilizadas por otro software.

A diferencia de un programa ejecutable, las librerías no cuentan con un punto de entrada principal, sino que su fin es ser utilizadas por otros programas independientes y de forma simultánea.

Con el fin de facilitar el desarrollo de la generación de códigos QR en el presente trabajo, se utilizará librerías para cada lenguaje de programación.

1.5.2 Librería de programación para el lenguaje Java

Primefaces

Es un popular marco de código abierto para JavaServer Faces que incluye más de 100 componentes, un kit de dispositivo móvil optimizado para el tacto, validación del lado del cliente, motor de temas y más.(PrimeTek, 2019)

Primefaces tiene los componentes necesarios para la generación del código QR, pero se necesitan de otras librerías complementarias para el cifrado de la información, cada una de ellas tiene una funcionalidad:

Barcode4j-light-2.1

es un generador flexible para códigos de barras escritos en Java.(MvnRepository, 2015)

core-3.3.3

Biblioteca de codificación / decodificación del núcleo del código de barras.

javase-3.3.0:Extensiones específicas de Java SE a la biblioteca ZXing⁴.(MvnRepository, 2018)

qrngen-1.4: Es un api de generación de QRCode simple para Java construido en ZXing.

1.5.3 Librería de programación para el lenguaje C#

MessagingToolkit. QRCode

QRCode Library es un componente C# (.NET) que se puede usar para codificar y decodificar códigos QR. Esta librería proporciona funciones para: codificar contenido en una imagen que puede guardarse en formatos JPEG, GIF, PNG o BITMAP.(Hellemons, 2012)

⁴ ZXing: Biblioteca de escaneo de código de barras ZXing ("Zebra Crossing") para Java, Android.

La biblioteca se puede utilizar en cualquier aplicación de Windows .NET 3.5, aplicaciones Web ASP.NET o aplicaciones de dispositivo Windows Mobile.(Hellemons, 2012)

1.5.4 Librería de programación para el lenguaje JavaScript

angularx-qrcode

Es una librería de componentes / módulos de Angular4-7.2 para generar códigos QR (Respuesta rápida) en su aplicación Angular 4/5/6/7.Esta librería se basa en la biblioteca qrcodejs.(Jacob, 2018)

1.5.5 Librería de programación para el lenguaje PHP

PHP QR code

Es una librería de código abierto (LGPL⁵) para generar código QR, código de barras bidimensional. Basado en la biblioteca libqrencode C, proporciona API para crear imágenes de código de barras de código QR (PNG, JPEG). Implementado exclusivamente en PHP, sin dependencias externas.(Karn, 2006)

Algunas de las características de la biblioteca incluyen:

- Soporta versiones de código QR (tamaño) 1-40
- Codificación numérica, alfanumérica, de 8 bits y Kanji.
- Implementado exclusivamente en PHP, sin dependencias externas excepto GD2
- Exporta a PNG, imágenes JPEG, también exporta como tabla de bits
- TCPDF 2-D API de integración de código de barras
- Fácil de configurar

1.6 ISO 25010

La ISO ha desarrollado estándares y modelos de software, con el objetivo de describir y evaluar la calidad de software. Hasta el 2005 la ISO 9126-1 era el modelo con el que se

⁵ LGPL: Licencia Publica General GNU Lesser, esta licencia garantiza la libertad de compartir y modificar el software autorizado por ella, asegurando que el software es libre para todos los usuarios.

trabajaba para calidad de software, esta fue remplazada por la ISO 25010 con la cual se trabaja en la actualidad.(Hussain & Mkpojiogu, 2015)

La ISO 25010 define la calidad de uso, como el grado en que un producto puede ser utilizado por usuarios específicos, observando si satisface sus necesidades para lograr objetivos definidos con eficacia, eficiencia, seguridad y satisfacción en contextos determinados.(Hussain & Mkpojiogu, 2015)

El modelo de calidad de la ISO/IEC 25010 se encuentra compuesta por ocho características específicas de calidad que se muestra a continuación.

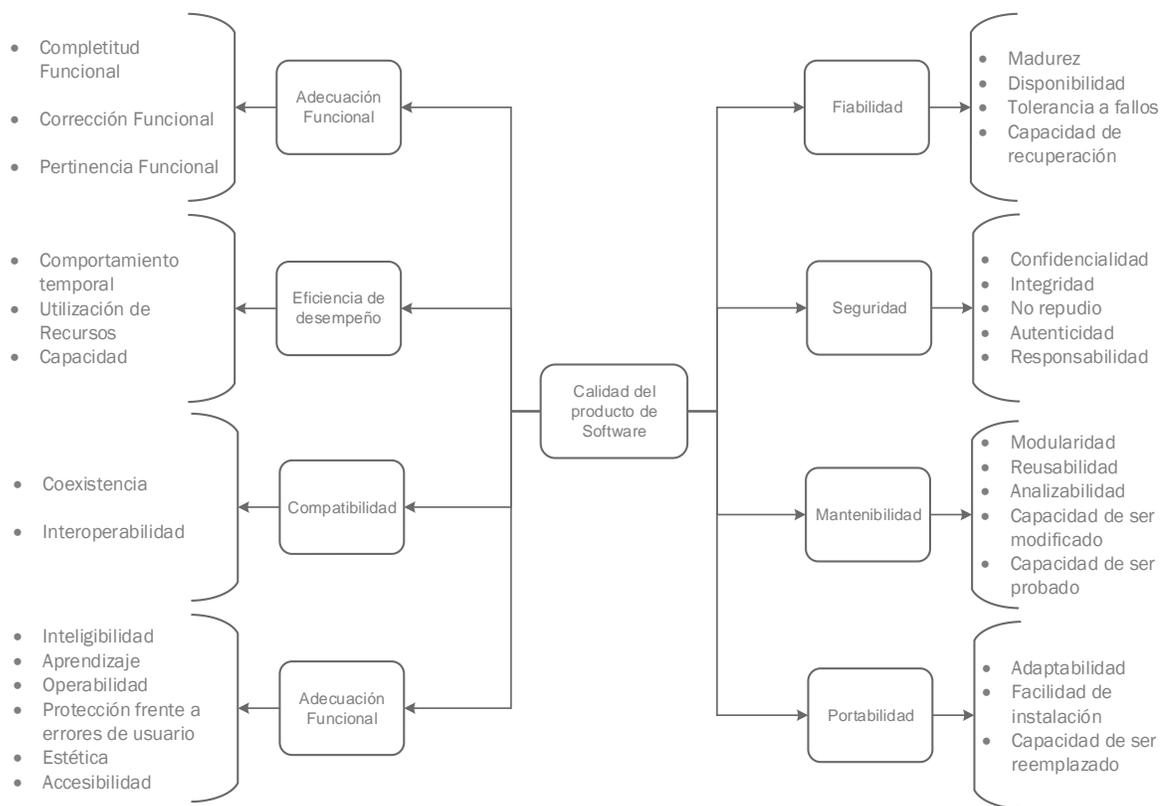


Figura. 7 Características de la ISO/IEC 25010
Fuente: (ISO, 2018)

Tomando en cuenta las características que define la ISO 25010, para analizar los complementos (librerías) que existen para generar códigos QR en los lenguajes de programación PHP, Java, JavaScript, solo se evaluará la característica de eficiencia de desempeño y sus respectivas sub-características.

- **Eficiencia de desempeño**

“Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones”(ISO, 2018). En la TABLA 6 se describe las sub-características de la eficiencia de desempeño:

TABLA 5 Sub-características de eficiencia de desempeño

SUB-CARACTERÍSTICA	DESCRIPCIÓN
Comportamiento temporal	Tiempo de respuesta que se demora el sistema para llevar a cabo sus funciones.
Utilización de recursos	Las cantidades y tipos de recursos utilizados cuando el software lleva a cabo su función bajo condiciones determinadas.
Capacidad.	Grado en que los límites máximos de un parámetro de un producto o sistema software cumplen con los requisitos.

Fuente: Propia

CAPÍTULO 2

2.1 Metodología de desarrollo

2.1.1 Modelo en cascada

Es un proceso de desarrollo secuencial, el software se lo realiza con un conjunto de etapas que se ejecutan una tras otra. Se le denomina así por las posiciones que ocupan las diferentes fases que componen el proyecto, colocadas una encima de otra, y siguiendo un flujo de ejecución de arriba hacia abajo, como una cascada.(Domínguez, 2017)

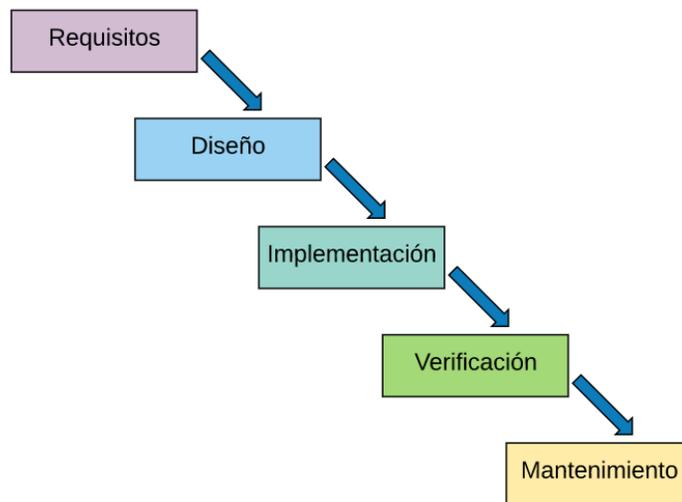


Figura. 8 Diseño del Modelo en Cascada.

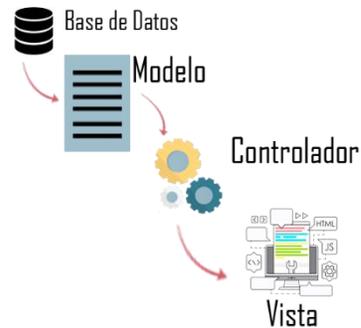
Fuente: <https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>

2.2 Requisitos

- Cada Demo realizado en los diferentes lenguajes de programación deben generar 1,10,100,500 y 1000 códigos QR.
- La información que se almacena en los códigos QR debe ser obtenida de una base de datos, la cual almacena información de los estudiantes de la carrera de Ingeniería en sistemas Computacionales.
- El código QR contendrá la cedula, nombres y apellidos de estos estudiantes.

2.3 Diseño

La implementación de los componentes de software se realizara en base al patrón de arquitectura MVC, esta arquitectura separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos (Modelo, Vista y Controlador).(Alicante, 2016)



2.4 Implementación de componentes de Software

Para la implementación de los componentes de software en cada lenguaje de programación, se trabajará con una base de datos **Postgres 9.6**, la cual tiene como nombre **CodigoQRTesis**, esta base de datos contiene una tabla llamada **tblestudiantes**, la cual contiene información de los estudiantes de la **Carrera de Ingeniería en sistemas computacionales** de la **Universidad Técnica del Norte**.

2.4.1 Implementación en el lenguaje Java

Para la implementación en este lenguaje de programación es necesario instalar:

- **WidFly 12:** Este es un servidor web que está diseñado específicamente para alojar aplicaciones Web desarrolladas en Java EE.
- **JSF 2.2:** Es una tecnología y framework para aplicaciones Java basadas en web. Este framework viene incluido cuando se instala el IDE Eclipse, específicamente la versión para desarrollo en Java EE.

Antes de empezar el desarrollo del Demo del aplicativo para la generación de código QR es necesario descargar las siguientes librerías:

- **Primefaces 7.0:** <https://www.primefaces.org/downloads/>

Esta es la librería principal, contiene los componentes para la generación del código QR, pero además se necesitan las siguientes librerías complementarias para el cifrado de información en el código QR:

- **Barcode4j-light-2.1:**
<https://mvnrepository.com/artifact/net.sf.barcode4j/barcode4j-light/2.1>
- **core-3.3.3:** <https://mvnrepository.com/artifact/com.google.zxing/core/3.3.3>
- **javase-3.3.0:** <https://mvnrepository.com/artifact/com.google.zxing/javase/3.3.0>
- **qrngen-1.4:** <https://mvnrepository.com/artifact/net.glxn/qrngen/1.4>

Una vez descargado todas las librerías necesarias, se realizará el desarrollo del demo para la generación del código QR, para este desarrollo se utilizará Eclipse Java EE 2018-09.

Para eso en el IDE Eclipse se crea un proyecto de tipo EAR Application Project. *Figura. 9*

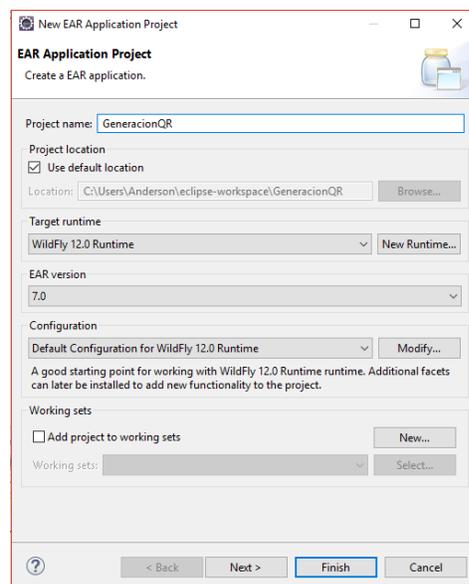


Figura. 9 Creación de un proyecto EAR Java (JSF)
Fuente: Propia.

Este proyecto contendrá dos carpetas, una EJB y otra WEB, La carpeta EJB contendrá el Modelo la aplicación, se alojará la conexión a la base de datos, Mapeo de información y scripts necesarios de la aplicación, la carpeta Web contendrá el Controller y el View de la aplicación.

Figura. 10

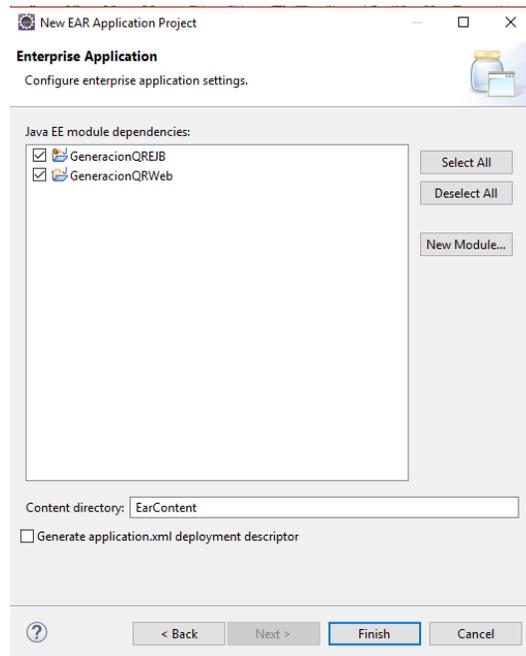


Figura. 10 Módulos en aplicaciones EAR Java (JSF)
Fuente: Propia.

2.4.1.1 Model

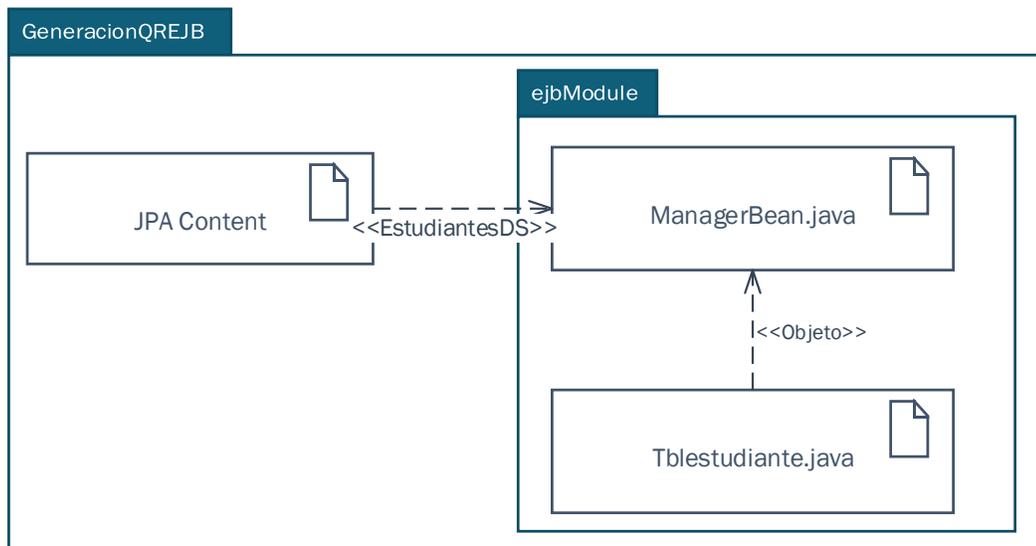


Figura. 11 Estructura del Módulo Model en Java (JSF).
Fuente: Propia

TABLA 6 Descripción de los elementos del módulo Model en Java (JSF)

Complemento	Descripción	Código
JPA Contente	En este complemento se configura la Persistencia o la conexión a la base de datos, este archivo está en formato xml (, se crea el método de conexión EstudiantesDS)	<pre><persistence version="2.1" xmlns="http://xmlns.jcp.org/xml/ns/persistence" xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:schemaLocation="http://xmlns.jcp.org/ xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/ persistence_2_1.xsd"> <persistence-unit name="EstudiantesDS"> <jta-data- source>java:/EstudiantesDS</jta-data-source> <class>model.Tblestudiante</class> </persistence-unit> </persistence></pre>
Tblestudiante.java	Este complemento sirve para el mapeo de información, es un objeto el cual contiene variables, las cuales representan a cada columna de la tabla la cual se hará el mapeo. Se debe implementar getters and setters para cada variable.	<pre>@Entity @Table(name="tblestudiantes") @NamedQuery(name="Tblestudiante.findAll", query="SELECT t FROM Tblestudiante t") public class Tblestudiante implements Serializable { private static final long serialVersionUID = 1L; @Id @Column(name="cedula_estudiante") private String cedulaEstudiante; @Column(name="apellido_estudiante") private String apellidoEstudiante; @Column(name="nombre_estudiante") private String nombreEstudiante;</pre>
ManagerBean.java	Se implementa un método el cual obtiene la información de la base de datos mediante código sql, y retorna una lista de objetos de tipo Estudiante. Se debe realizar la Persistencia a la base de datos, para poder hacer el mapeo de la información.	<pre>@Stateless @LocalBean public class ManagerBean { @PersistenceContext(unitName = "EstudiantesDS") private EntityManager em; @SuppressWarnings("unchecked") public List<Tblestudiante> listaEstudiantes(){ Query q = em.createQuery("select u from Tblestudiante u"); q.setMaxResults(1); return q.getResultList(); } }</pre>

Fuente: Propia

2.4.1.2 Controller

Para el desarrollo del Controller en este demo en las propiedades de la carpeta Web en la opción Project Faces se instala el framework JSF 2.2, el cual ayudará a cumplir los objetivos de esta parte del desarrollo.

Es necesario que el módulo Controller tenga comunicación con el módulo Model, en este caso la comunicación de la carpeta WEB con la EJB, para eso se agrega al módulo EJB en las propiedades del módulo Web en la opción Build Patch.

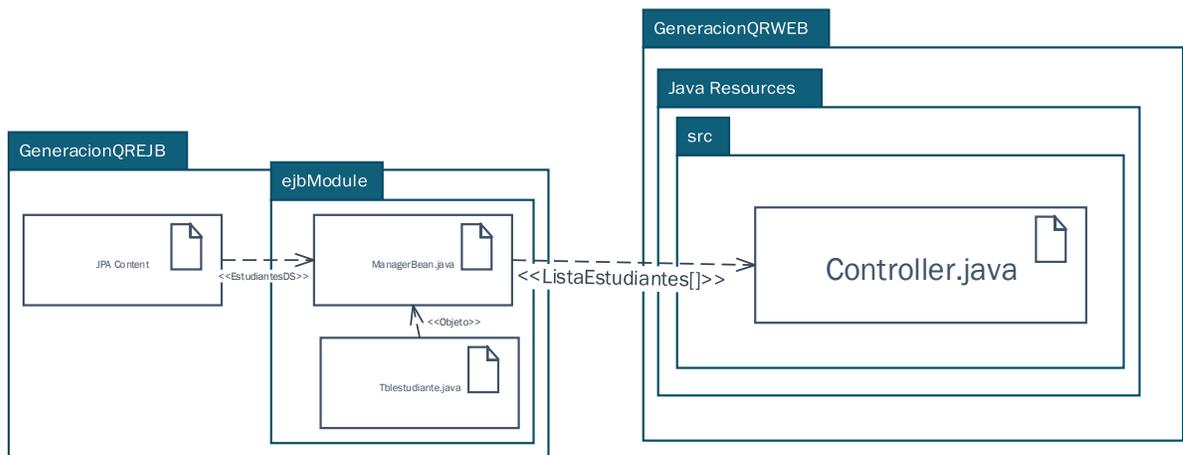


Figura. 12 Estructura del Módulo Controller en Java (JSF)
Fuente: Propia

TABLA 7 Descripción de los elementos del módulo Controller en Java (JSF)

Complemento	Descripción	Código
Controller.java	En este complemento se obtiene la información que retorna el EJB del proyecto, esta información se almacena en una lista y se sube a una Session, con el objetivo que esta información sea manipulable en el navegador.	<pre> @SessionScoped @ManagedBean public class Controller { List<Tblestudiante> estudiantes; @EJB ManagerBean managerestudiante; @PostConstruct public void iniciar() { estudiantes= managerestudiante.listaEstudiantes(); } public List<Tblestudiante> getEstudiante1() { return estudiante; } public void setEstudiante1(List<Tblestudiante> estudiante) { this.estudiante = estudiante; } } </pre>

Fuente: Propia

2.4.1.3 View

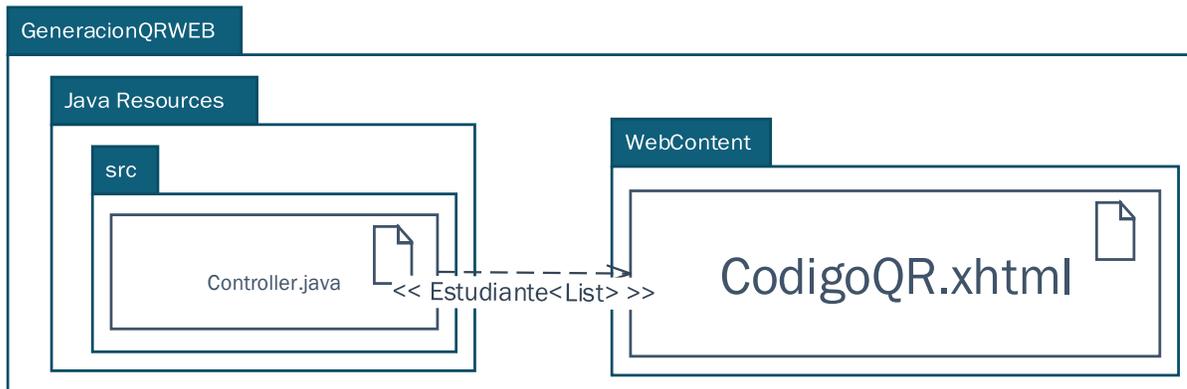


Figura. 13 Estructura del Módulo View en Java (JSF)
Fuente: Propia

TABLA 8 Descripción de los elementos del módulo View en Java (JSF)

Complemento	Descripción	Código
CodigoQR.xhtml	<p>Este complemento es el que muestra los resultados en el navegador, para esto se obtiene la información subida a la Session. En este caso la lista estudiante, con esta información se genera los códigos QR, la codificación se realiza con la librería Primefaces 7.0.</p> <p>En la <i>Figura. 14</i> se presenta el proceso que lleva generar un código QR con la librería Primefaces 7.0.</p>	<pre><html xmlns="http://www.w3.org/1999/xhtml" xmlns:ui="http://java.sun.com/jsf/facelets" xmlns:f="http://java.sun.com/jsf/core" xmlns:h="http://java.sun.com/jsf/html" xmlns:p="http://primefaces.org/ui"> <h:head></h:head> <body> <h:outputText value="#{controller.obtenerhora()}" /> <p:dataTable value="#{controller.estudiante1000}" var="estudiante"> <p:column > <h:form id="formQR"> <p:barcode value="#{estudiante.nombreEstudiante}" type="qr" width="25%" height="25%" align="center"></p:barcode> </h:form> </p:column> </p:dataTable> <h:outputText value="#{controller.obtenerhora()}" /> </body> </html></pre>

Fuente: Propia

Proceso de generación de un código QR con la librería Primefaces 7.0.

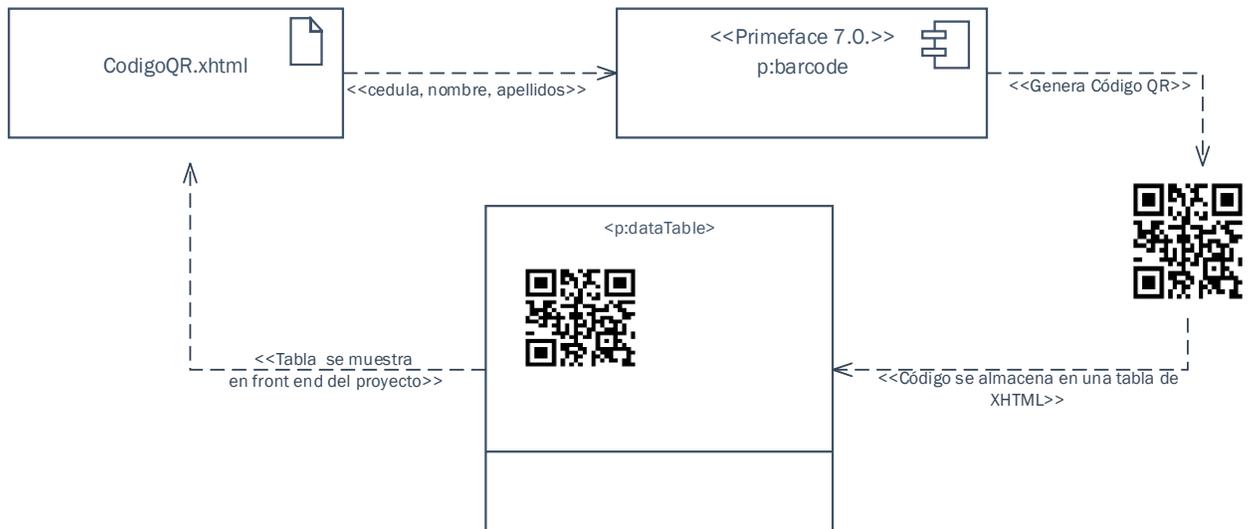


Figura. 14 Proceso de generación de un código QR con la librería Primefaces 7.0
Fuente: Propia

2.4.2 Implementación en el lenguaje C#

Para la implementación en este lenguaje de programación es necesario tener instalado

- **ASP .Net:** es un marco web de código abierto para crear aplicaciones y servicios web modernos con C# (.NET).(Microsoft, 2018)

Para empezar, se crea una nueva Aplicación Web ASP .NET, en la configuración del proyecto se escoge la opción MVC, ya que esta es el patrón de arquitectura que se está utilizando en este trabajo.

En el proyecto se crean todas las carpetas necesarias incluidas las de la arquitectura MVC, en este caso Models, Views y Controllers, en estas carpetas se desarrollará el Demo para generar códigos QR con el lenguaje de programación C#. *Figura. 15*

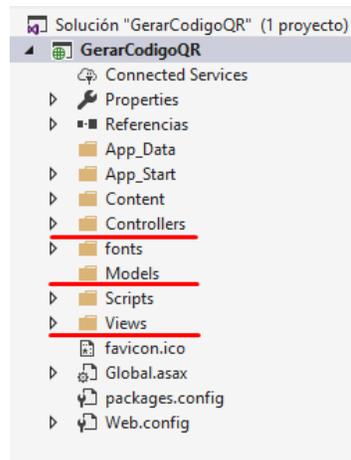


Figura. 15 Proyecto creado en C#
Fuente: Propia

Es necesario descarga las siguientes librerías:

- **EntityFramework6.Npgsql v3.2.2:** Es un proveedor de datos de fuente abierta ADO.NET para PostgreSQL, permite que los programas escritos en C #, Visual Basic, F # accedan al servidor de base de datos PostgreSQL. Se implementa en código 100% C #, es gratuito y es de código abierto.(Npgsql, 2018)
- **MessagingToolkit.QRCode v1.3.0:** QRCode Library es un componente .NET que se puede usar para codificar y decodificar QRCode.

Estas librerías se instalan con la ayuda del administrador Nuget, este es el gestor de paquetes para .NET. Las herramientas cliente de NuGet proporcionan la capacidad de producir y consumir paquetes creados en .NET.(Nuget, 2018) Esta galería se la puede encontrar haciendo click derecho en el proyecto y ahí se encuentra la opción **Administrador Nuget**.

Una vez instalado los paquetes se realizará la configuración para la conexión a la base de datos, para eso en el archivo **Web.config** del proyecto se agrega las siguientes líneas de código, preferentemente al último del archivo.

```
<system.data>
  <DbProviderFactories>
    <add name="Npgsql Data Provider" invariant="Npgsql" description=".Net Data Provider for PostgreSQL" type="Npgsql.NpgsqlFactory, Npgsql" support="FF" />
```

```

</DbProviderFactories>
</system.data>

<connectionStrings>
  <add name="postgres" providerName="Npgsql"
  connectionString="Server=localhost;DataBase=Nombre de base de datos;User
  ID=postgres;Pwd=Contraseña" />
</connectionStrings>

```

2.4.2.1 Model

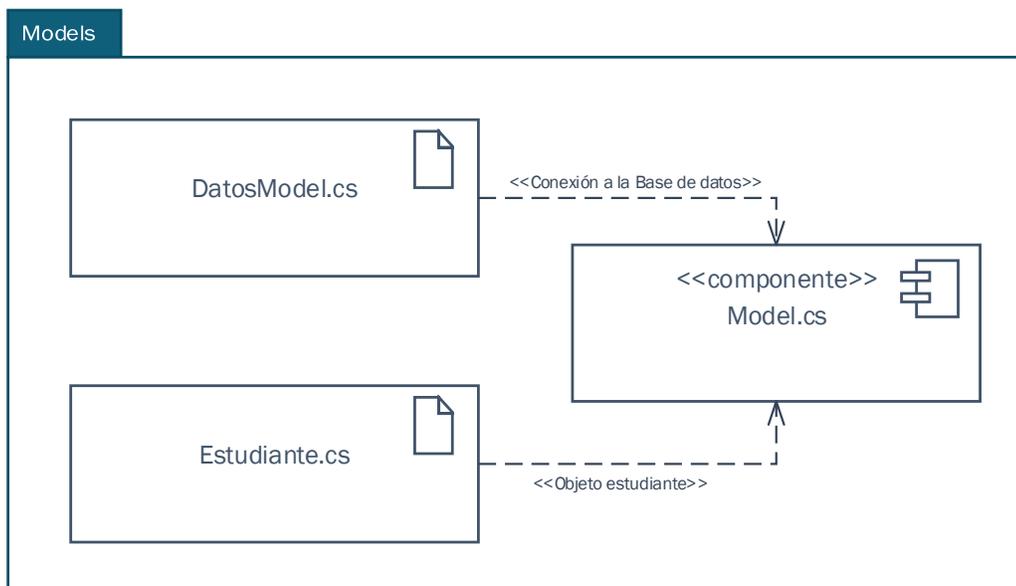


Figura. 16 Estructura del Módulo Models en C# (.NET)
Fuente: Propia

TABLA 9 Descripción de los elementos del módulo Models en C# (.NET)

Complemento	Descripción	Código
DatosModel.cs	Contienen un método el cual hace referencia a la base de datos y a la tabla con la que se está trabajando. En este método se utiliza la configuración hecha anteriormente en el archivo Web.config .	<pre> public class DatosModel:DbContext { public DatosModel() : base("postgres") {} public DbSet<Estudiante> Estudiantes { get; set; } </pre>
Estudiante.cs	Este objeto es crea variables en base a los campos que contiene la base de datos, por tal razón se debe especificar el nombre de la tabla con la que se va a trabajar, y el nombre de las variables tienen que ser igual a los campos de esta tabla.	<pre> [Table("tblestudiantes", Schema = "public")] public class Estudiante { [Key] public String cedula_estudiante { get; set; } public String nombre_estudiante { get; set; } public String apellido_estudiante { get; set; } public String celular_estudiante { get; set; } public String correo_estudiante { get; set; } } </pre>

Model.cs	Este archivo contiene métodos los cuales sirven para obtener la información de la base de datos y almacenarlos en un array de objetos de tipo <i>Estudiante</i> .	<pre>DatosModel conex = new DatosModel(); public List<Estudiante> Estudiantes() { var lista = conex.Estudiantes.SqlQuery("select * from tblestudiantes") .ToList<Estudiante>(); return lista; }</pre>
-----------------	---	---

Fuente: Propia

2.4.2.2 Controller

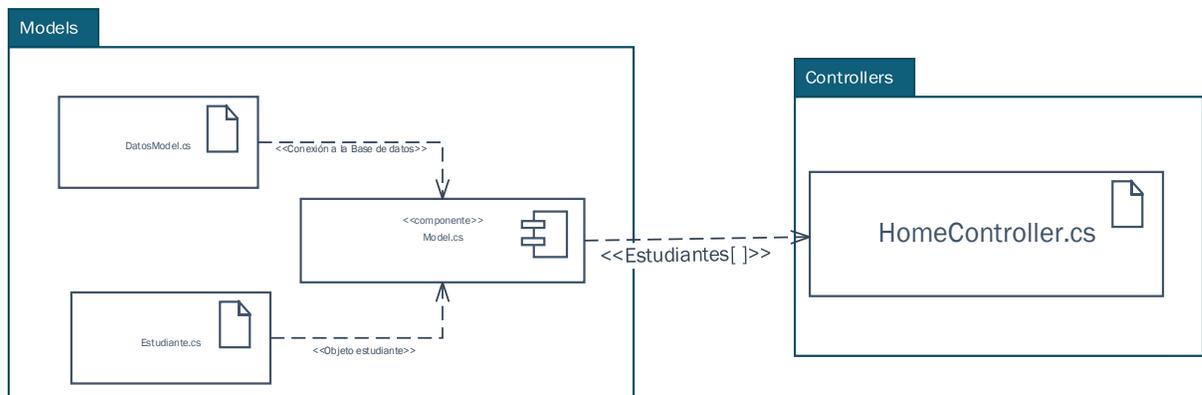


Figura. 17 Estructura del Módulo Controller en C# (.NET)

Fuente: Propia

TABLA 10 Descripción de los elementos del módulo Controller en C# (.NET)

Complemento	Descripción	Código
Controller.cs	Este archivo obtiene la información o el array de objetos obtenidos de la base de datos, esta información es enviada a la vista del proyecto. Se consume el componente Model.cs programado anteriormente en el módulo Model.	<pre>public ActionResult CodigosQR() { Model mod = new Model(); DatosModel _context; _context = new DatosModel(); ViewBag.Estudiantes= mod.obtenerEstudiantes().ToList(); return View(); }</pre>

Fuente: Propia

2.4.2.3 View

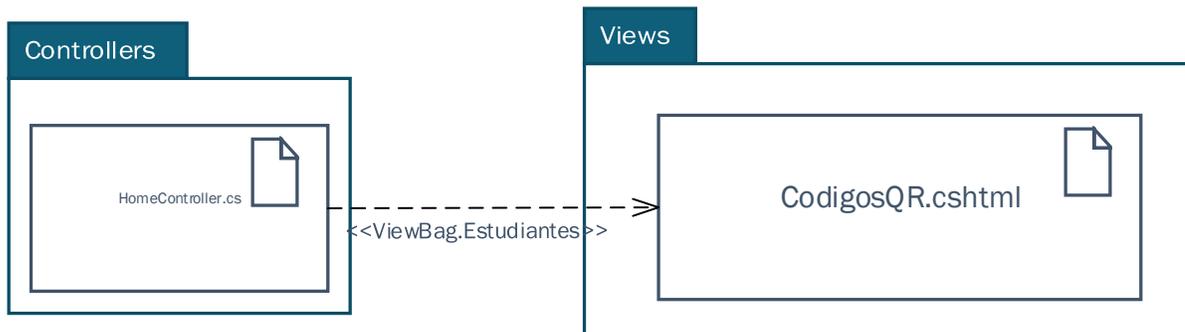


Figura. 18 Estructura del Módulo View en C# (.NET)
Fuente: Propia

TABLA 11 Descripción de los elementos del módulo View en C# (.NET)

Complemento	Descripción	Código
CodigosQR.cshtml	<p>Este archivo consume el array de objetos tipo estudiante enviado por el Controller, obtiene la información y la envía a codificar, y a su vez lo muestra en pantalla.</p> <p>En la <i>Figura.19</i> se muestra el proceso de generación de un código QR con la librería MessagingToolkit.QRCode v1.3.0.</p>	<pre>@using MessagingToolkit.QRCode.Codec; @using MessagingToolkit.QRCode.Codec.Data; @using System.Drawing; @using System.Drawing.Imaging; @{ ViewBag.Title = "CodigosQR"; var Estudiantes = ViewBag.Estudian1; int count = 0; } @foreach (var est in Estudiantes) { QRCodeEncoder encoder = new QRCodeEncoder(); Bitmap img = encoder.Encode(est.cedula_estudiante + est.nombre_estudiante + est.apellido_estudiante); img.Save("C:\\Users\\Anderson\\source\\repos\\ GerarCodigoQR\\GerarCodigoQR\\Controllers\\Imagenes\\img" + count + ".jpg", ImageFormat.Jpeg); string imagen = "img" + count + ".jpg";

 count++; } }</pre>

Fuente: Propia

Proceso de generación de un código QR con la librería MessagingToolkit.QRCode v1.3.0.

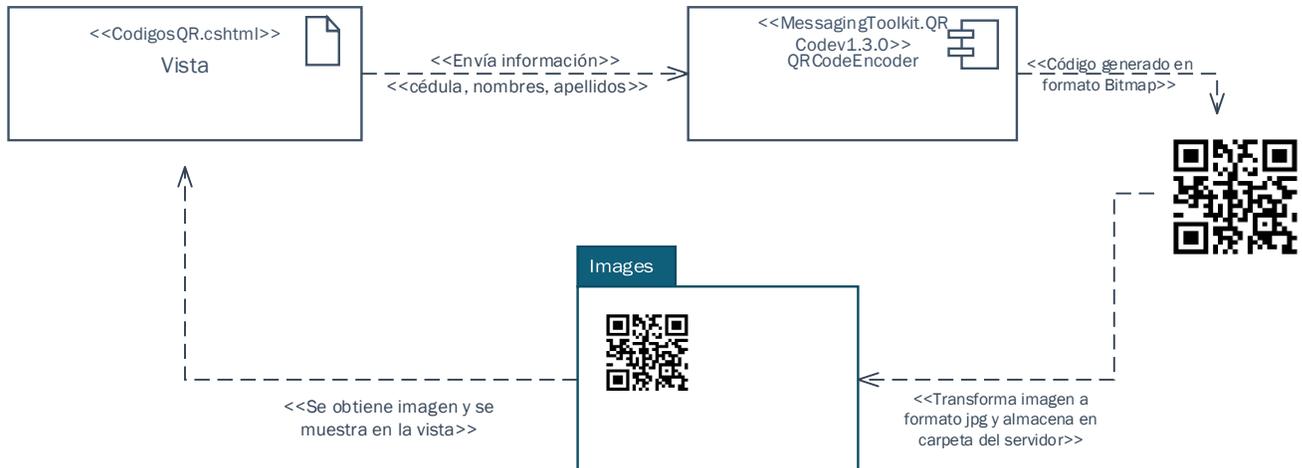


Figura. 19 Proceso de generación de un código QR con la librería MessagingToolkitQRCode Fuente: Propia

2.4.3 Implementación en el lenguaje Java Script

Para la implementación en este lenguaje de programación es necesario tener instalado:

- **Node.js:** Es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node está diseñado para construir aplicaciones en red escalables.(Foundation, 2018)

Este entorno se puede descargar en el siguiente enlace:

<https://nodejs.org/es/download/>

- **Angular 7:** Es una plataforma que facilita la creación de aplicaciones web. Angular combina plantillas declarativas, inyección de dependencia, herramientas de extremo a extremo y mejores prácticas, integradas para resolver los desafíos de desarrollo.(Google, 2019)

Para la configuración y descarga de esta plataforma se puede ingresar al siguiente enlace:

<https://angular.io/guide/quickstart>

- **Express js Framework:** Express es una infraestructura de aplicaciones web Node.js mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles. (StrongLoop, s/f)

La instalación de este Framework se lo realiza mediante consola.

Código de instalación: `npm install -g express-generator`

El desarrollo de este demo tendrá dos etapas, una etapa de estas es un proyecto que sirve para la obtención de datos mediante el *framework express*, esta información se mostrará en un archivo JSON, el cual será consumido por la aplicación front end del Framework Angular. De esta manera se estaría cumpliendo el modelo de arquitectura MVC.

2.4.3.1 Model

Creación del proyecto Express mediante la siguiente línea de consola: `express (Nombre de carpeta)`. *Figura. 20*

```
Anderson@DESKTOP-3EMEEIH MINGW64 /c/Tesis Demos
$ express backendQR
warning: the default view engine will not be jade in future releases
warning: use '--view=jade' or '--help' for additional options
```

Figura. 20 Creación de carpeta Express Nodejs
Fuente: Propia

Para el desarrollo es necesario descargar los siguientes complementos de programación.

TABLA 12 Complementos de programación JavaScript (Express)

Complemento	Descripción	Código de instalación
Node js	Entorno de desarrollo node js	<code>npm install</code>
pg	Conexión a la base de datos postgres	<code>npm install --save pg</code>
pg-promise	Conexión a la base de datos postgres	<code>npm install --save pg-promise</code>
bluebird	Generador de archivos JSON	<code>npm install --save bluebird</code>

Fuente: Propia

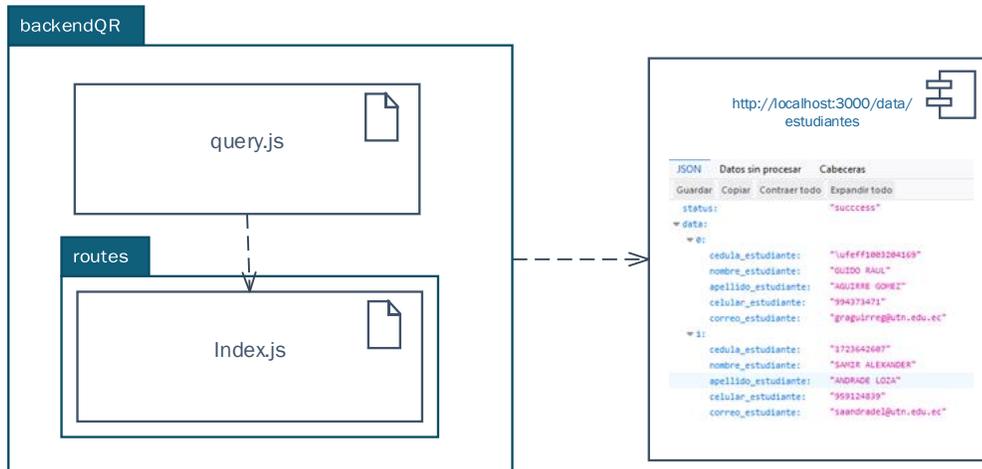


Figura. 21 Estructura del Módulo Model con JavaScript (Express)
Fuente: Propia

TABLA 13 Descripción de los elementos del módulo Model en JavaScript (Express)

Complemento	Descripción	Código
query.js	<p>Contiene una consulta script la cual obtiene la información de la base de datos en este caso de la tabla estudiantes. La información obtenida se almacena en un archivo JSON.</p>	<pre>var promise = require('bluebird'); var options = { promiseLib: promise } var pgp = require('pg-promise')(options) var connectionString = 'postgres://postgres:12345@localhost:5432/CodigoQRTesis'; var db = pgp(connectionString); function getList(req, res, next) { db.any("SELECT * FROM public.tblestudiantes") .then((data) => { res.status(200) .json({ status: 'success', data: data, message: 'Retrived list' }); }).catch((err) => { return next(err); }); } module.exports = { getList: getList, }</pre>
Index.js	<p>La información almacenada en el archivo JSON tiene que ser consumida mediante una ruta de navegación, en este archivo se configura la ruta mediante la cual será consumida. En este caso: http://localhost:3000/data/estudiantes</p>	<pre>var express = require('express'); var router = express.Router(); var db = require('../query'); router.get('/data/estudiantes',db.getList); module.exports = router;</pre>

Fuente: Propia

2.4.3.2 Controller

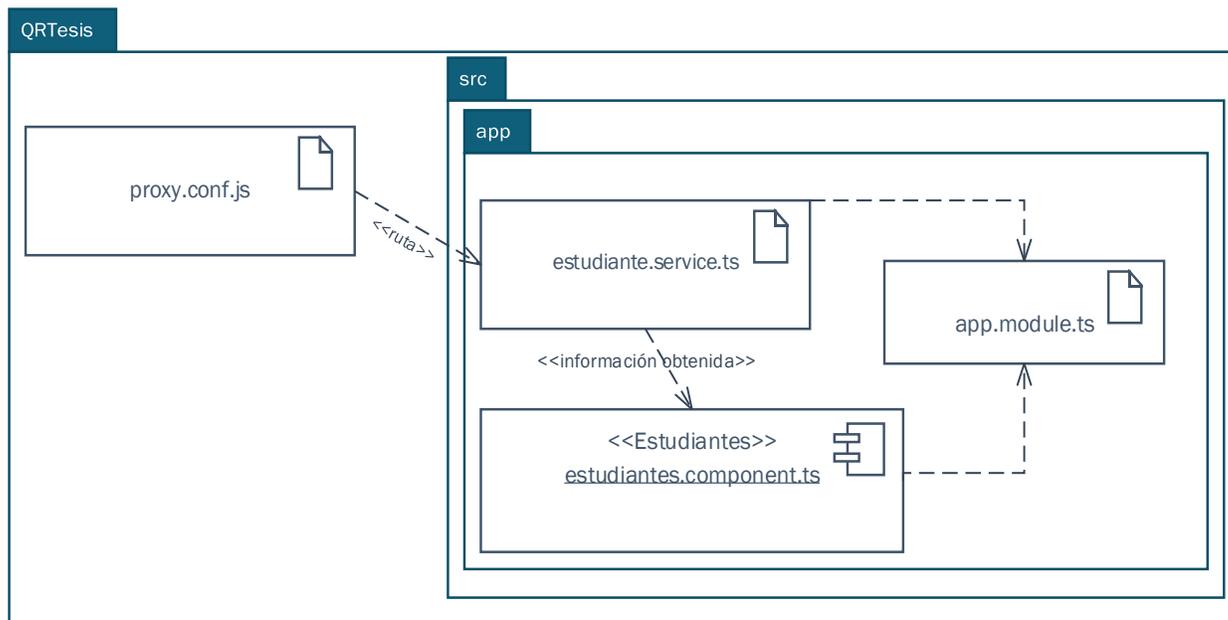


Figura. 22 Estructura del Módulo Controller con JavaScript (Angular)
Fuente: Propia

TABLA 14 Descripción de los elementos del módulo Controller en JavaScript (Angular)

Complemento	Descripción	Código
Proxy.conf.js	Para acceder a la información presentada en formato Json es necesario crear este archivo el cual contiene la configuración de la ruta de la cual se va a obtener los datos.	<pre>{ "/data/*": { "target": "http://localhost:3000", "secure":false, "changeOrigin":true } }</pre>
estudiante.service.ts	Con este servicio se hace conexión a la ruta del archivo Json, se crea un método el cual retorna la información almacenada en el archivo. Código de generación de un servicio: <i>ng generate service estudiante</i>	<pre>import { Injectable } from '@angular/core'; import { HttpClient } from '@angular/common/http'; import { Observable } from 'rxjs'; @Injectable({ providedIn: 'root' }) export class EstudianteService { constructor(private httpClient: HttpClient) {} obtenerEstudiantes(): Observable<any> { return this.httpClient.get("/data/estudiantes"); } }</pre>

estudiantes.component.ts	<p>Se hace el llamado al método el cual se creó para a hacer la conexión a la ruta del archivo Json, se obtiene esta información y se almacena en una variable. Código de generación de un componente: <i>ng g c estudiantes</i></p>	<pre>@Component({ selector: 'app-quinientosestudiantes', templateUrl: './quinientosestudiantes.component.html', styleUrls: ['./quinientosestudiantes.component.css'] }) export class QuinientosestudiantesComponent implements OnInit { fecha: any; fecha2: any; estudiantes: any; constructor(private estudianteService: EstudianteService) { this.obtenerEstudiante(); } obtenerEstudiante() { this.estudianteService.obtener500Estudiantes() .subscribe(resultado => { this.estudiantes = resultado.data; }, error => { console.log(JSON.stringify(error)); }); } ngOnInit() { } }</pre>
app.module.ts	<p>Todos los objetos de programación creados en el proyecto se deben centralizar en este módulo, con el fin que puedan ser utilizados por otros módulos u objetos.</p>	<pre>import { EstudianteService } from './estudiante.service'; import { EstudianteComponent } from './estudiante/estudiante.component'; @NgModule({ declarations: [EstudianteComponent,], providers: [EstudianteService], })</pre>

Fuente: Propia

2.4.3.3 View

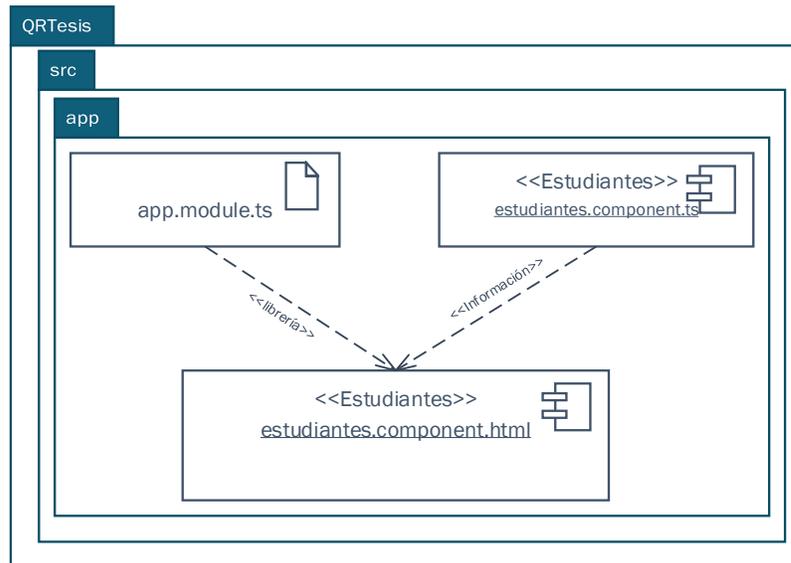


Figura. 23 Estructura del Módulo View con JavaScript (Angular)
Fuente: Propia

TABLA 15 Descripción de los elementos del módulo View en JavaScript

Complemento	Descripción	Código
app.module.ts	Importación de la librería <i>angularx-qrcode</i> , la cual sirve para codificación de información a código QR.	<pre>import { QRCodeModule } from 'angularx-qrcode'; @NgModule({ imports: [QRCodeModule], })</pre>
estudiante.service.html	<p>Consumes la información obtenida de la base de datos y hace la generación de códigos QR con la librería <i>angularx-qrcode</i>.</p> <p>A continuación, en la <i>Figura.24</i> se muestra el proceso de generación de un código QR con esta librería.</p>	<pre><div class="containter"> <div class="row"> <div class="col-8"> <table class="table table-light"> <thead class="thead-light"> <tr> <th>Codigo QR</th> </tr> </thead> <tbody> <tr *ngFor="let estudiante of estudiantes"> <td><qrcode [qrdata]= estudiante.cedula_estudiante+ estudiante.nombre_estudiante+ estudiante.apellido_estudiante [size]="256" [level]="M"></qrcode></td> </tr> </tbody> </table> </div> </div> </div></pre>

Fuente: Propia

Proceso de generación de un código QR con la librería angularx-qrcode.

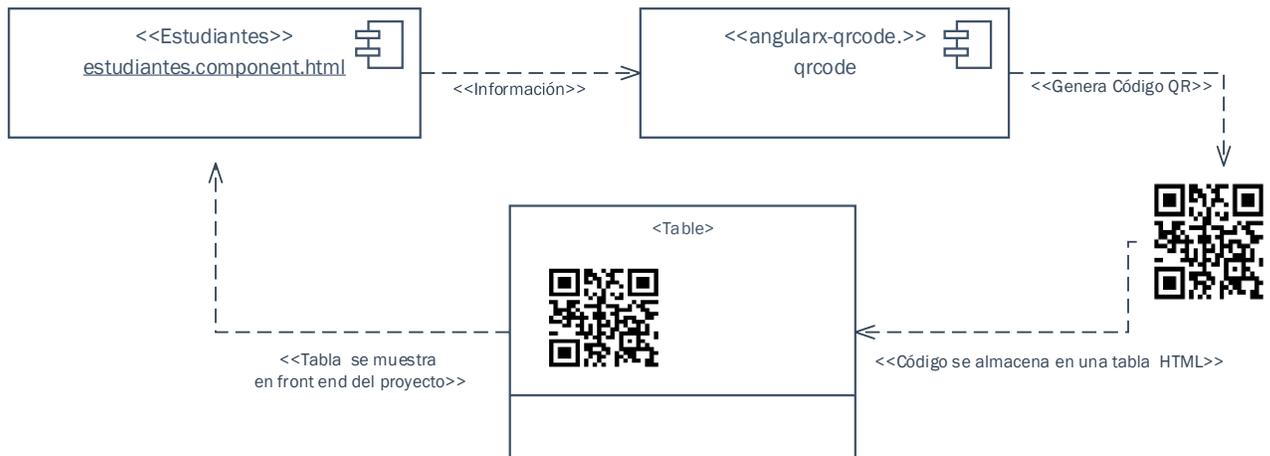


Figura. 24 Proceso de generación de un código QR con la librería angularx-qrcode
Fuente: Propia

2.4.4 Implementación en el lenguaje PHP

Para la implementación de los componentes de software en PHP es necesario tener un escenario compuesto por:

- **Servidor XAMPP v3.2.2:** Servidor el cual servirá para alojar la aplicación Web.
- **PHP 7.2.12:** Lenguaje de programación con el cual se trabajará.

Antes de Implementar los componentes de software es necesario descargar la librería **Phpqrcode versión 2010100721_1.14**, la cual servirá para generar códigos QR en este lenguaje de programación. Esta librería se podrá descargar en el siguiente enlace:

<https://sourceforge.net/projects/phpqrcode/files/releases/>

Para el desarrollo del Demo en el lenguaje php. Se crea una carpeta en el servidor XAMPP, xampp->htdocs, esta carpeta almacena subcarpetas las cuales contendrán complementos de programación que son útiles para el funcionamiento de este Demo, estas se crean en base al patrón de arquitectura utilizada (MVC). *Figura.9*

La carpeta descargada de la librería **Phpqrcode**, debe ser almacena en esta misma ruta del proyecto.

Nombre	Fecha de modif...	Tipo
controller	26/11/2018 19:55	Carpeta de archivos
model	26/11/2018 19:38	Carpeta de archivos
phpqrcode	26/11/2018 20:33	Carpeta de archivos
view	09/01/2019 20:19	Carpeta de archivos

Figura. 25 Carpetas creadas en el proyecto PHP
Fuente: Propia

2.4.4.1 Model

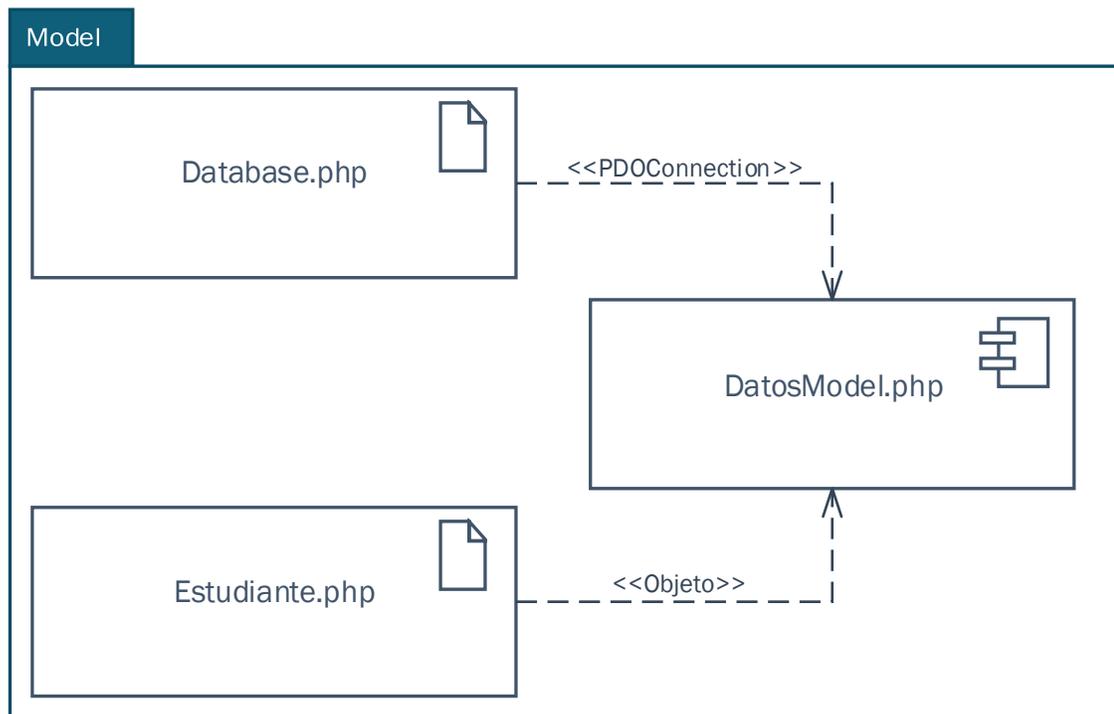


Figura. 26 Estructura del Módulo Model en php.
Fuente: Propia

TABLA 16 Descripción de los elementos del Módulo Model en php

Complemento	Descripción	Código
Database.php	Se implementa métodos de programación los cuales contienen código que sirve para hacer la conexión de la aplicación con la base de datos.	<pre>self::\$conexion=new PDO (" pgsql : host =" .self::\$dbHost.";"dbname=". self::\$dbName, self::\$dbUsername, self::\$dbUserPassword);</pre>
Estudiante.php	Es un objeto el cual contienen variables y un constructor que sirve para hacer el mapeo de información.	<pre>function_construct(\$cedulaEstudiante, \$nombres, \$apellidos, \$celular, \$correo) { \$this->cedulaEstudiante = cedulaEstudiante; \$this->nombres = \$nombres; \$this->apellidos = \$apellidos; \$this->celular = \$celular; \$this->correo = \$correo;</pre>

DatosModel.php	Este archivo contiene métodos de programación los cuales ayudan a obtener la información de la base de datos, esto se puede realizar con la ayuda de los dos archivos creados anteriormente. Se crea una variable la cual contiene la conexión a la base de datos y se hace una consulta sql la cual obtiene la información, la misma que es almacenada en un array de objetos de tipo Estudiante.	<pre>include '../model/Database.php'; include_once '../model/Estudiante.php'; public function ObtenerEstudiante() { \$pdo = Database::connect(); \$sql = "select * from tblestudiantes"; \$resultado = \$pdo->query(\$sql); foreach (\$resultado as \$res) { \$estudiante = new Estudiante(\$res['cedula_estudiante'], \$res['nombre_estudiante'], \$res['apellido_estudiante'], \$res['celular_estudiante'], \$res['correo_estudiante']); array_push(\$listado, \$estudiante);}}</pre>
-----------------------	---	---

Fuente: Propia

2.4.4.2 Controller

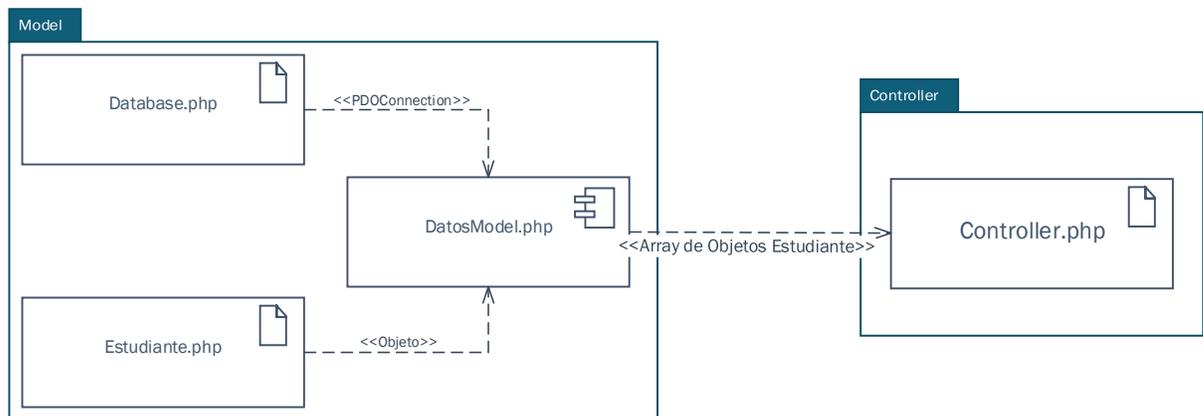


Figura. 27 Estructura del Módulo Controller en php.

Fuente: Propia

TABLA 17 Descripción de los elementos del Módulo Controller en php

Complemento	Descripción	Código
Controller.php	Este archivo contiene métodos de programación los cuales obtienen la información mapeada por el módulo Model, en esta etapa la información que retorna del Model se sube a una SESSION con el objetivo que esta sea manipulable en el módulo View.	<pre>session_start(); include_once '../model/DatosModel.php'; \$datosModel = new DatosModel(); \$listado = \$datosModel->ObtenerEstudiante(); \$_SESSION['estudiantes'] = serialize(\$listado);</pre>

Fuente: Propia

2.4.4.3 View

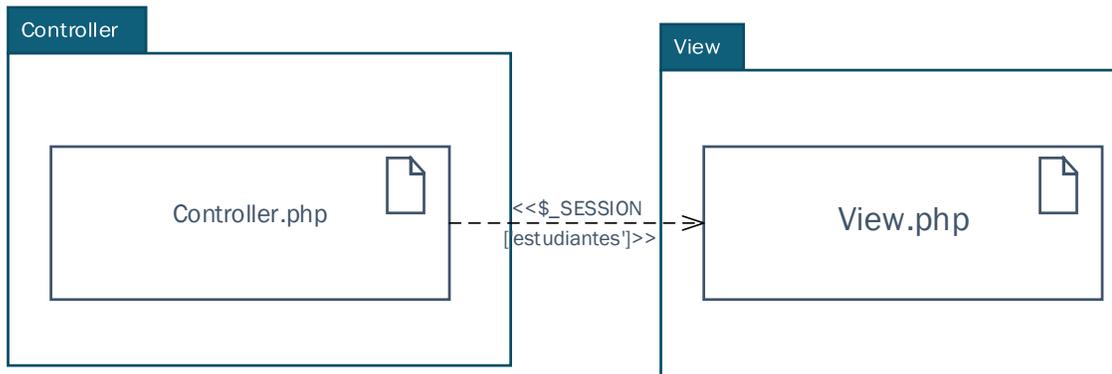


Figura. 28 Estructura del Módulo View en php.
Fuente: Propia

TABLA 18 Descripción de los elementos del Módulo View en php

Complemento	Descripción	Código
View.php	<p>Este archivo consume la SESSION creada en el controller del proyecto, codifica la información obtenida en códigos QR y se presentan en pantalla.</p> <p>En la <i>Figura.29</i> se presenta el proceso que lleva la generación de un código QR con la librería Phpqrcode</p>	<pre> session_start(); require '../phpqrcode/qrlib.php'; include_once '../model/Estudiante.php'; \$dir='temp/'; if(!file_exists(\$dir)) mkdir(\$dir); \$contador=1; if (isset(\$_SESSION['estudiantes'])) { \$contenido=unserialize(\$_SESSION['estudiantes']); foreach (\$contenido as \$cont) { \$filename= \$dir.'test'.\$contador.'.png'; \$tamano=5; \$level='H'; \$frameSize=6; \$conteni=\$cont->getCedulaEstudiante().\$cont->getNombres().\$cont->getApellidos(); QRCode::png(\$conteni, \$filename, \$level, \$tamano, \$frameSize); echo ''; \$contador++; } } </pre>

Fuente: Propia

Proceso de generación de un código QR con la librería PhpqrCode

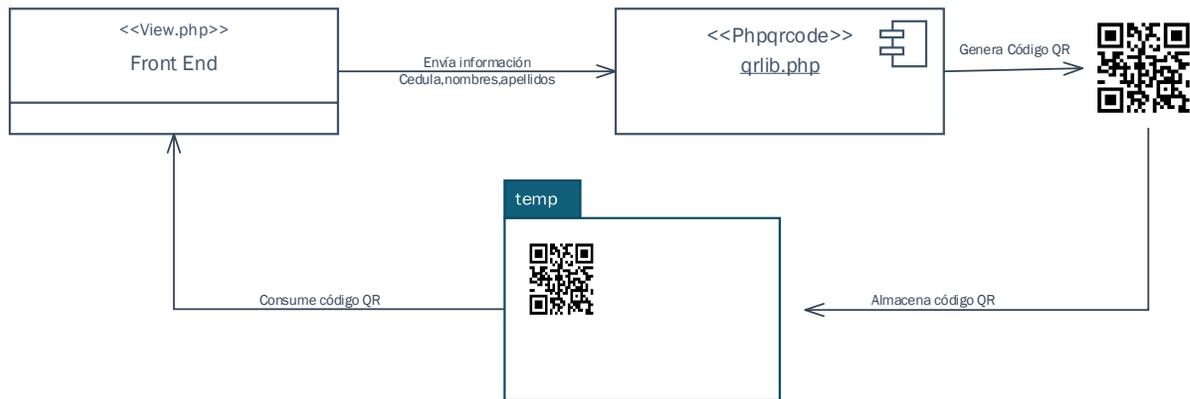


Figura. 29 Proceso de generación de un código QR con la librería PhpqrCode.
Fuente: Propia

2.5 Comparativa en base a la característica 2 de la ISO 25010

Esta característica trata sobre la eficiencia y rendimiento de un producto de software, el rendimiento es relativo a la cantidad de recursos usados en bajo las condiciones establecidas.

Las condiciones que se utilizará en este trabajo serán, que cada Demo desarrollado genere 1 código QR, 10 códigos QR, 100 códigos QR, 500 códigos QR y 1000 códigos QR. Esto ayudara a obtener información medible sobre el rendimiento de cada lenguaje de programación al momento de generar códigos QR.

2.5.1 Comportamiento de tiempo

Grado en el cual la respuesta, los tiempos de procesamiento y las tasas de rendimiento de un producto o sistema satisfacen los requisitos al desempeñar sus funciones.

a) Tiempo de respuesta

Métrica: Tiempo de respuesta

Propósito: ¿Cuál es el tiempo estimado para completar una tarea?

Método: Tomar el tiempo desde que se envía la petición hasta obtener la respuesta.

Variables: *A*: Tiempo de envío de petición. *B*: Tiempo en recibir la respuesta.

Formula: $X = B - A$.

b) Tiempo de espera

Métrica: Tiempo de espera.

Propósito: ¿Cuál es el tiempo desde que se envía una instrucción, para que inicie un trabajo, hasta que lo completa?

Método: Tomar el tiempo cuando se inicia un trabajo y el tiempo en completar el trabajo.

Variables: *A*: Tiempo cuando se inicia un trabajo. *B*: Tiempo en completar el trabajo.

Formula: $X = B - A$.

c) Rendimiento

Métrica: Rendimiento.

Propósito: ¿Cuántas tareas pueden ser procesadas por unidad de tiempo?

Método: Contar el número de tareas completadas en un intervalo de tiempo.

Variables: *A*: Tiempo cuando se inicia un trabajo. *T*: Intervalo de tiempo. Donde $T > 0,000$

Formula: $X = A / T$.

2.5.2 Utilización de recursos

Grado en el cual las cantidades y tipos de recursos usados por un producto o sistema satisfacen los requisitos al desempeñar sus funciones.

a) Utilización de CPU

Métrica: Utilización de CPU

Propósito: ¿Cuánto tiempo de CPU es usado para realizar una tarea dada?

Método: Tomar el tiempo de operación y la cantidad de tiempo de CPU que se usa para realizar una tarea.

Variables: *A*: La cantidad de tiempo de CPU que realmente es usado para realizar una tarea. *B*: Tiempo de operación.

Formula: $X = B - A$.

b) Utilización de la memoria

Métrica: Utilización de la memoria.

Propósito: ¿Cuánto espacio de memoria es usado para realizar una tarea dada?

Método: Medir la cantidad total de espacios de memoria y la cantidad de espacios de memoria que realmente es usado para realizar una tarea

Variables: *A*: Cantidad de espacios de memoria que realmente es usado para realizar una tarea. *B*: Cantidad total de espacios de memoria.

Formula: $X = B - A$.

c) Utilización de los dispositivos de E/S

Métrica: Utilización de los dispositivos de E/S

Propósito: ¿Cuánto tiempo los dispositivos de E/S utilizan para realizar una tarea?

Método: Tomar el tiempo de operación y el tiempo que los dispositivos de E/S pasan ocupados para realizar la tarea

En este caso esta métrica no se aplica, ya que no se utiliza elementos de Entrada y Salida al momento de generar un código, no obstante, se podría utilizar al momento de imprimir un código QR, pero no es el propósito de este trabajo.

2.5.3 Capacidad

Grado en el cual los límites máximos de un parámetro de un producto o sistema satisface los requisitos.

a) Número de peticiones online

Métrica: Número de peticiones online.

Propósito: ¿Cuántas peticiones online pueden ser procesadas por unidad de tiempo?

Método: Contar el número máximo de peticiones online procesadas y tomar el tiempo de operación.

Variables: A : Número máximo de peticiones online procesada. T = Tiempo de operación

Formula: $X = A / T$.

b) Número de accesos simultáneos

Métrica: Número de accesos simultáneos.

Propósito: ¿Cuántos usuarios pueden acceder al sistema simultáneamente en un cierto tiempo?

Método: Contar el número máximo de accesos simultáneos y tomar el tiempo de operación

Variables: A : Número máximo de accesos simultáneos. T = Tiempo de operación

Formula: $X = A / T$

c) Sistema de transmisión de ancho de banda

Métrica: Sistema de transmisión de ancho de banda.

Propósito: ¿Cuánto es el valor límite absoluto de transmisión necesaria para cumplir con las funciones?

Método: Contar la cantidad máxima de transmisión de datos y tomar el tiempo de operación.

Variables: A : Cantidad máxima de transmisión de datos. T = Tiempo de operación

Formula: $X = A / T$

2.6 Análisis de implementación

Al haber finalizado la implementación de los componentes de software en los diferentes lenguajes de programación se encontró algunas ventajas y desventajas en diferentes puntos de la implementación.

TABLA 19 Ventajas y desventajas de implementación.

	VENTAJAS	DESVENTAJAS
JAVA	<ul style="list-style-type: none"> -Existe mucha información sobre la instalación y uso de este lenguaje de programación. -La librería que se utiliza para la generación de los códigos QR tiene su información muy bien centralizada en una página web. 	<ul style="list-style-type: none"> -La librería que se utiliza para general los códigos QR utiliza librerías complementarias.
JavaScript	<ul style="list-style-type: none"> -La información sobre la instalación y uso de los complementos de programación de este lenguaje esta centralizado en páginas web. -El uso de la librería para la generación de los códigos QR es fácil de utilizar. 	<ul style="list-style-type: none"> -La forma de implementación de la librería para la generación QR cambia dependiendo de la versión que se utilice.
C#	<ul style="list-style-type: none"> -Este lenguaje de programación es fácil de instalar. -La librería que se utiliza para la generación de los códigos QR es fácil de utilizar. 	<ul style="list-style-type: none"> -No existe información suficiente sobre la utilización de este lenguaje de programación.
PHP	<ul style="list-style-type: none"> -Este lenguaje de programación es fácil de instalar. -Es un lenguaje de programación fácil de utilizar. Existe mucha información sobre este lenguaje. -La librería que se utiliza para generar los códigos QR es fácil de implementar 	<ul style="list-style-type: none"> -No es un lenguaje de programación seguro.

Fuente: Propia

CAPÍTULO 3

Resultados

3.1 Pruebas

A continuación, se muestra las pruebas realizadas a cada lenguaje de programación, las pruebas se realizaron en base a las métricas establecidas anteriormente sobre la ISO-25010. Se utilizó la herramienta *J-Meter*⁶ y *Monitor de Recursos*⁷ para obtener los datos resultantes de las pruebas.

3.1.1 Comportamiento de tiempo

Los datos que se presentan en las pruebas de *Tiempo de Respuesta* y *Tiempo de Espera*, se presentan con el siguiente formato: **ss,ms** donde, **ss** representa los segundos de un tiempo determinado y **ms** representan los milisegundos. Para las pruebas de *Rendimiento* los datos son de tipo **número**.

a) Tiempos de Respuesta

Como se puede apreciar en la *Figura. 30*, los valores de tiempo para los lenguajes Java y Javascript no tienen cambios significativos, a diferencia C# y PHP, los valores de estos experimentan grandes cambios de acuerdo con la cantidad de códigos QR que se generan.

⁶ J-Meter: Es una herramienta que está diseñada para cargar el comportamiento funcional de un software y medir el rendimiento de este.

⁷ Monitor de Recursos: es una herramienta que viene preinstalada en el sistema operativo Windows, la cual sirve para medir la utilización de recursos en procesos específicos.

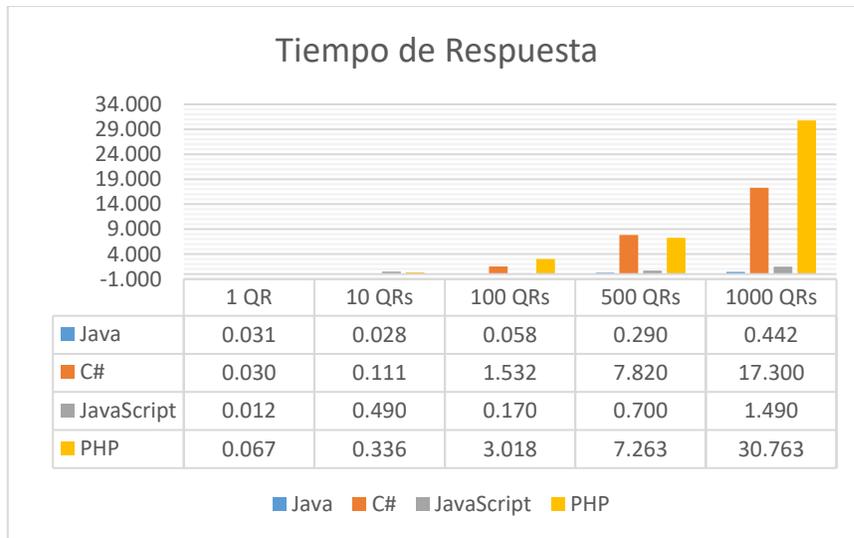


Figura. 30 Tiempos de Espera en los diferentes Lenguajes de Programación.
Fuente: Propia

b) Tiempos de Espera

Como se puede apreciar en la *Figura. 31*, los valores de tiempo para el lenguaje Java no tiene cambios significativos, a diferencia C#, JavaScript y PHP, los valores de estos experimentan grandes cambios de acuerdo con la cantidad de códigos QR que se generan.

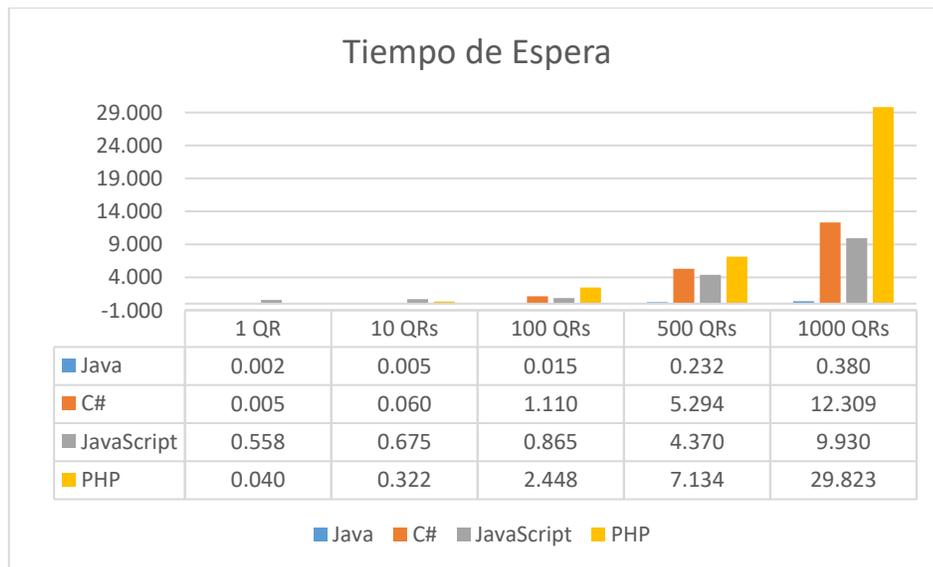


Figura. 31 Tiempos de Respuesta en los diferentes Lenguajes de Programación.
Fuente: Propia

c) Rendimiento

Como se puede observar en la *Figura .32*, los lenguajes PHP y C# tienen mejor rendimiento cuando se generan cantidades bajas de códigos QR, sin embargo, cuando el número de códigos sube, el rendimiento tiene una caída significativa, a diferencia de Java y JavaScript, los resultados no cambian drásticamente.

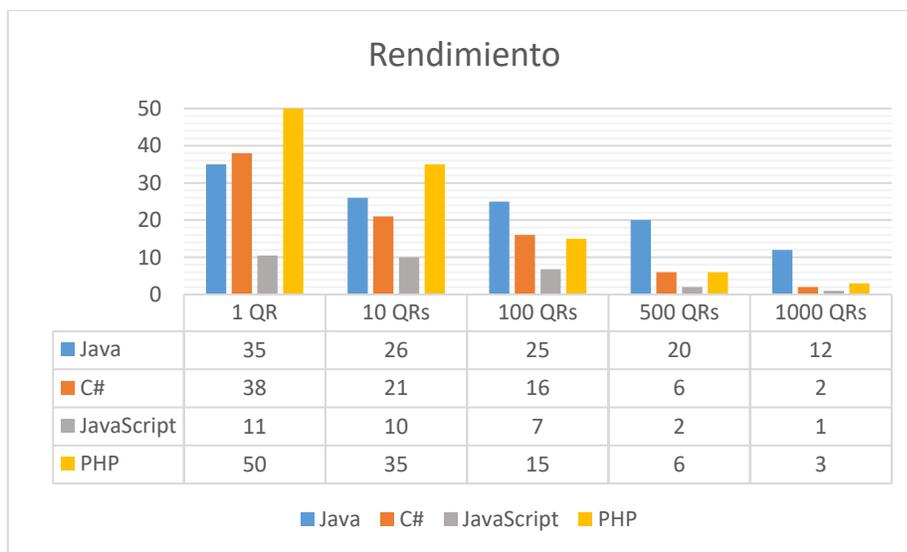


Figura. 32 Datos de Rendimiento en los diferentes Lenguajes de Programación.
Fuente: Propia.

3.1.2 Utilización de recursos.

Los datos que se presentan en la prueba *Utilización de CPU*, se presenta con el siguiente formato: **ss,ms** donde, **ss** representa los segundos de un tiempo determinado y **ms** representan los milisegundos. Para las pruebas de *Utilización de Memoria* los datos son de tipo: unidades de información, en este caso representan **Megabytes**.

a) Utilización de CPU

Como se puede apreciar en la *Figura. 33* el Lenguaje C# tiene cambios significativos en los tiempos de utilización de CPU, a diferencia de Java, JavaScript y PHP, los valores de estos no experimentan grandes cambios de acuerdo con la cantidad de códigos QR que se generan.

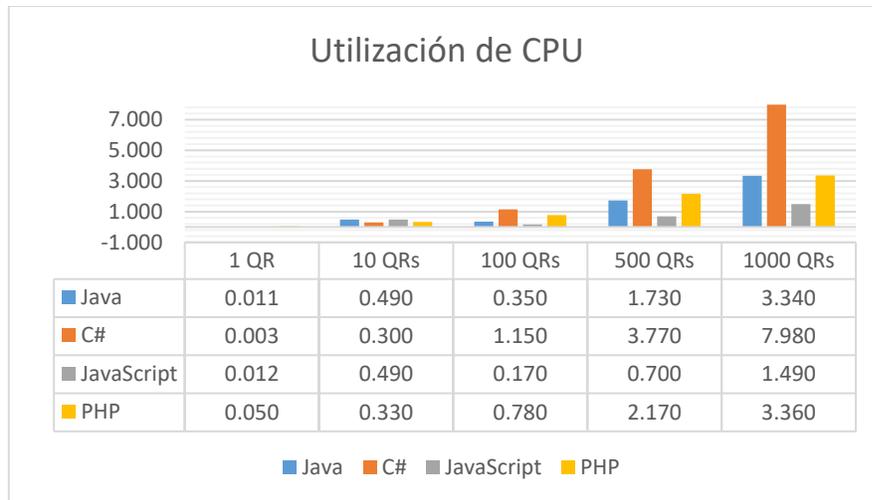


Figura. 33 Utilización de CPU en los diferentes lenguajes de Programación.
Fuente: Propia.

b) Utilización de memoria

Como se puede apreciar en la *Figura. 34*, el Lenguaje C# tiene cambios significativos en los tiempos de utilización de CPU, a diferencia de Java, JavaScript y PHP, los valores de estos no experimentan grandes cambios de acuerdo con la cantidad de códigos QR que se generan.

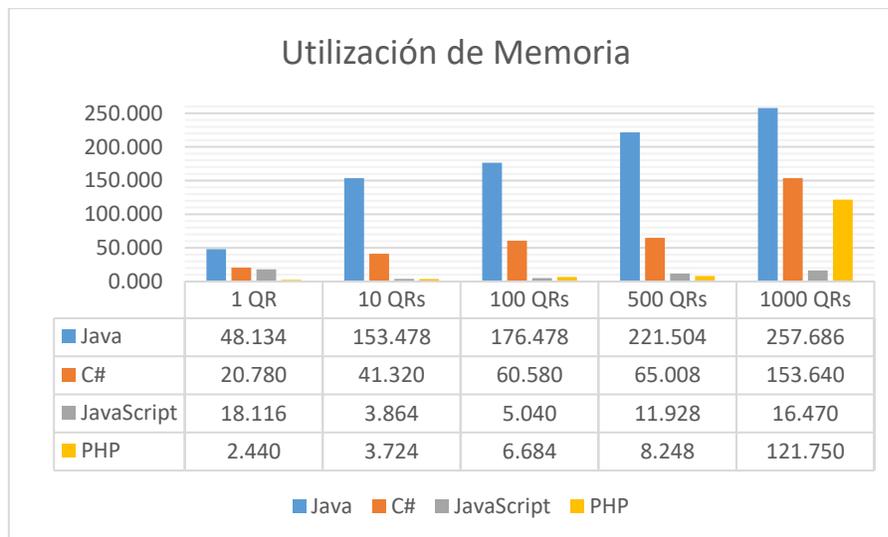


Figura. 34 Utilización de memoria en los diferentes Lenguajes de Programación
Fuente: Propia.

3.1.3 Capacidad.

Los datos que se presentan en las pruebas de *Números de Peticiones Online* y *Números de accesos simultáneos* son de tipo **número**. Para las pruebas de *Sistema de transmisión de ancho de banda* los datos son de tipo: unidades de información, en este caso representan **Megabytes**.

a) Números de Peticiones Online

Como se puede apreciar en la Figura. 35, los valores para los lenguajes Java, JavaScript y PHP no tiene cambios significativos, a diferencia C#, los valores de este experimentan grandes cambios de acuerdo con la cantidad de códigos QR que se generan.

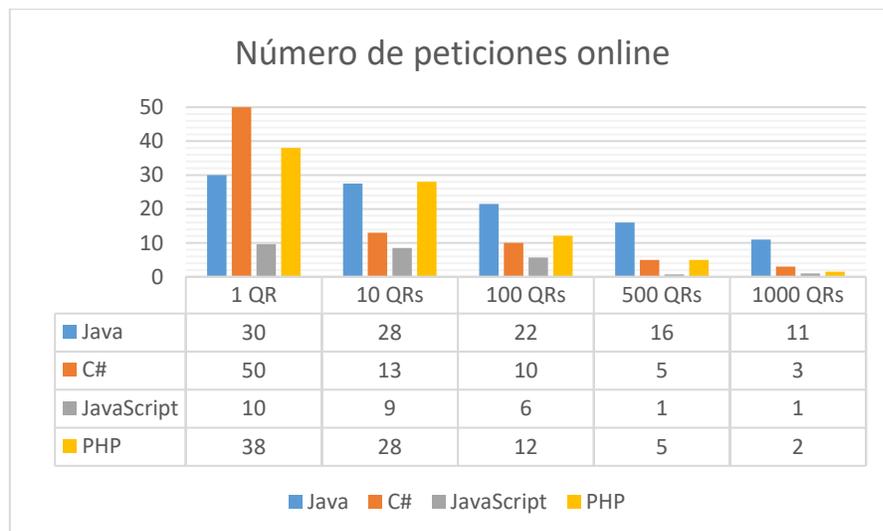


Figura. 35 Número de peticiones online en los diferente Lenguajes de Programación
Fuente: Propia.

b) Número de accesos simultáneos

Como se puede apreciar en la *Figura. 36*, los números para los lenguajes Java y JavaScript no tiene cambios significativos, a diferencia C# y PHP, los números de estos experimentan grandes cambios de acuerdo con la cantidad de códigos QR que se generan.

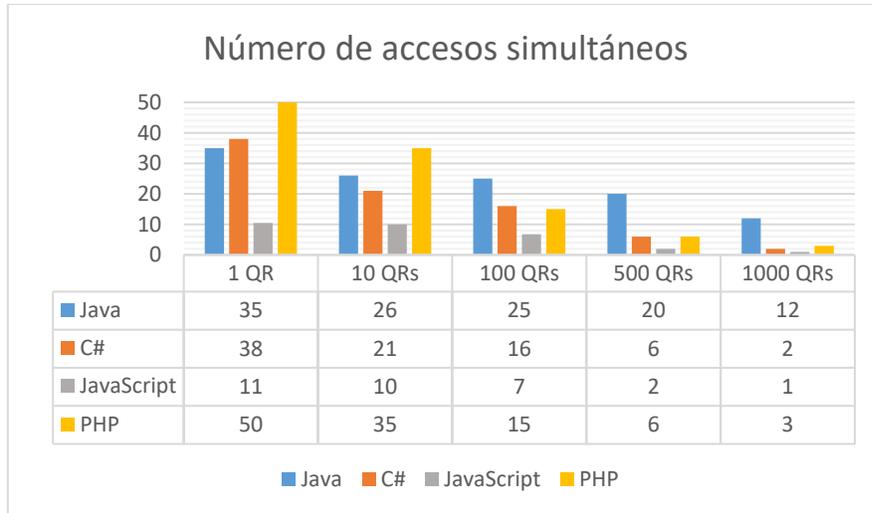


Figura. 36 Número de accesos simultáneos en los diferentes Lenguajes de Programación
Fuente: Propia.

c) Sistema de transmisión de ancho de banda

Como se puede apreciar en la *Figura. 37*, el ancho de banda para los lenguajes JavaScript y PHP es bajo y no tiene cambios significativos, a diferencia Java y C#, las medidas de estos experimentan grandes cambios de acuerdo con la cantidad de códigos QR que se generan.

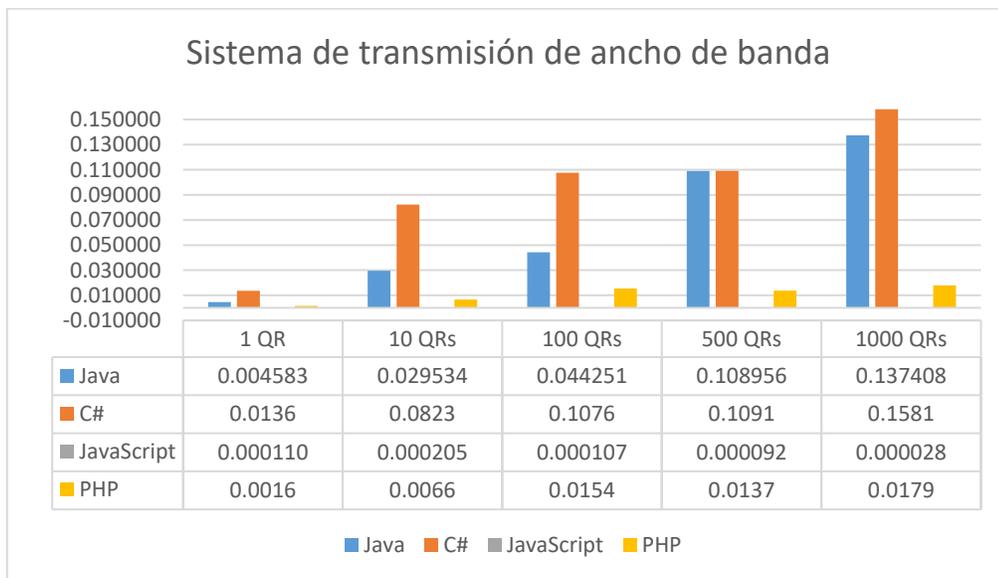


Figura. 37 Sistema de transmisión de ancho de banda en los diferentes Lenguajes de Programación.
Fuente: Propia.

3.2 Análisis de Resultados

Con los resultados obtenidos de las pruebas realizadas en base a métricas específicas, se creó una matriz en la cual se muestra los valores de: Peor caso y Valor deseado, con el fin de realizar una evaluación a cada lenguaje de programación y obtener una calificación que tendrá formato “10/10”. Las pruebas y la matriz se implementaron en base a la característica 2 de la ISO 25010.

Para la evaluación en la Matriz, se eligieron los datos de las pruebas realizadas con el número de códigos QR = 1000.

a) Lenguaje de programación Java

Características	Subcaracterísticas	Métrica	Peor caso	Valor Deseado	Aplica	Variables			Valor Obtenido	Valor Métrica / 10	Final Subcaracterísticas	Total Características
						A	B	T	X			
Eficiencia en el desempeño	Comportamiento del tiempo	Tiempo de respuesta	> 20,000 seg	<= 10,000 seg	Si	4,909	5,352		0,442	10,00	7,50	9,94
		Tiempo de espera	> 16,000 seg	<= 8,000seg	Si	24,124	24,504		0,380	10,00		
		Rendimiento	0 tareas	>= 5 tareas	Si	120		10,000	12	10,00		
	Utilización de recursos	Utilización de CPU	> 8 seg	<=4seg	Si	0,420	3,76		3,340	10,00	0,94	
		Utilización de la memoria	> 280 mb	<= 140 mb	Si	96,54	354,23		257,69	8,82		
		Utilización de los dispositivos de E/S			No					0,00		
	Capacidad	Número de peticiones online	0 peticiones	>= 4 peticiones	Si	110,00		10,000	11	10,00	1,50	
		Número de accesos simultáneos	0 accesos	>= 5 accesos	Si	120,00		10,000	12	10,00		
		Sistema de transmisión de ancho de banda	> 1mb	<=1mb	Si	1,27		28,70	0,0443	10,00		

Figura. 38 Evaluación de eficiencia de desempeño con el lenguaje Java.

Fuente: Propia

b) Lenguaje de programación C#

Características	Subcaracterísticas	Métrica	Peor caso	Valor Deseado	Aplica	Variables			Valor Obtenido	Valor Métrica / 10	Final Subcaracterísticas	Total Características
						A	B	T	X			
Eficiencia en el desempeño	Comportamiento del tiempo	Tiempo de respuesta	> 20,000 seg	<= 10,000 seg	Si	30,691	47,992		17,300	7,30	5,40	7,25
		Tiempo de espera	> 16,000 seg	<= 8,000seg	Si	28,423	40,733		12,309	6,31		
		Rendimiento	0 tareas	>= 5 tareas	Si	20		10,000	2	8,00		
	Utilización de recursos	Utilización de CPU	> 8 seg	<=4seg	Si	0,530	8,51		7,980	2,02	0,59	
		Utilización de la memoria	> 280 mb	<= 140 mb	Si	120,25	273,89		153,64	9,86		
		Utilización de los dispositivos de E/S			No					0,00		
	Capacidad	Número de peticiones online	0 peticiones	>= 4 peticiones	Si	30,00		10,000	3	7,00	1,25	
		Número de accesos simultáneos	0 accesos	>= 5 accesos	Si	20,00		10,000	2	8,00		
		Sistema de transmisión de ancho de banda	> 1mb	<=1mb	Si	5,82		53,36	0,1091	10,00		

Figura. 39 Evaluación de eficiencia de desempeño con el lenguaje C#.

Fuente: Propia

c) Lenguaje de programación JavaScript

Características	Subcaracterísticas	Métrica	Peor caso	Valor Deseado	Aplica	Variables			Valor Obtenido	Valor Métrica / 10	Final Subcaracterísticas	Total Características
						A	B	T	X			
Eficiencia en el desempeño	Comportamiento del tiempo	Tiempo de respuesta	> 20,000 seg	<= 10,000 seg	Si	3,886	14,354		10,467	0,47	3,35	5,77
		Tiempo de espera	> 16,000 seg	<= 8,000seg	Si	18,624	28,554		9,930	3,93		
		Rendimiento	0 tareas	>= 5 tareas	Si	10		10,000	1	9,00		
	Utilización de recursos	Utilización de CPU	> 8 seg	<=4seg	Si	0,010	1,50		1,490	10,00	1,00	
		Utilización de la memoria	> 280 mb	<= 140 mb	Si	22,09	38,56		16,47	10,00		
		Utilización de los dispositivos de E/S			No					0,00		
	Capacidad	Número de peticiones online	0 peticiones	>= 4 peticiones	Si	5,00		10,000	1	9,50	1,43	
		Número de accesos simultáneos	0 accesos	>= 5 accesos	Si	10,00		10,000	1	9,00		
		Sistema de transmisión de ancho de banda	> 1mb	<=1mb	Si	0,00		30,11	0,0000	10,00		

Figura. 40 Evaluación de eficiencia de desempeño con el lenguaje JavaScript.

Fuente: Propia

d) Lenguaje de programación PHP

Características	Subcaracterísticas	Métrica	Peor caso	Valor Deseado	Aplica	Variables			Valor Obtenido	Valor Métrica / 10	Final Subcaracterística	Total Característica
						A	B	T				
Eficiencia en el desempeño	Comportamiento del tiempo	Tiempo de respuesta	> 20,000 seg	<= 10,000 seg	Si	12,429	43,192		30,763	0,00	1,75	4,03
		Tiempo de espera	> 16,000 seg	< =8,000seg	Si	13,650	43,474		29,823	0,00		
		Rendimiento	0 tareas	> = 5 tareas	Si	30		10,000	3	7,00		
	Utilización de recursos	Utilización de CPU	> 8 seg	<=4seg	Si	0,050	3,41		3,360	10,00	1,00	
		Utilización de la memoria	> 280 mb	<= 140 mb	Si	224,86	346,61		121,75	10,00		
		Utilización de los dispositivos de E/S			No					0,00		
	Capacidad	Número de peticiones online	0 peticiones	>= 4 peticiones	Si	15,00		10,000	2	8,50	1,28	
		Número de accesos simultáneos	0 accesos	>= 5 accesos	Si	30,00		10,000	3	7,00		
		Sistema de transmisión de ancho de banda	> 1mb	<=1mb	Si	0,84		46,98	0,0179	10,00		

Figura. 41 Evaluación de eficiencia de desempeño con el lenguaje PHP.

Fuente: Propia

- Resultado

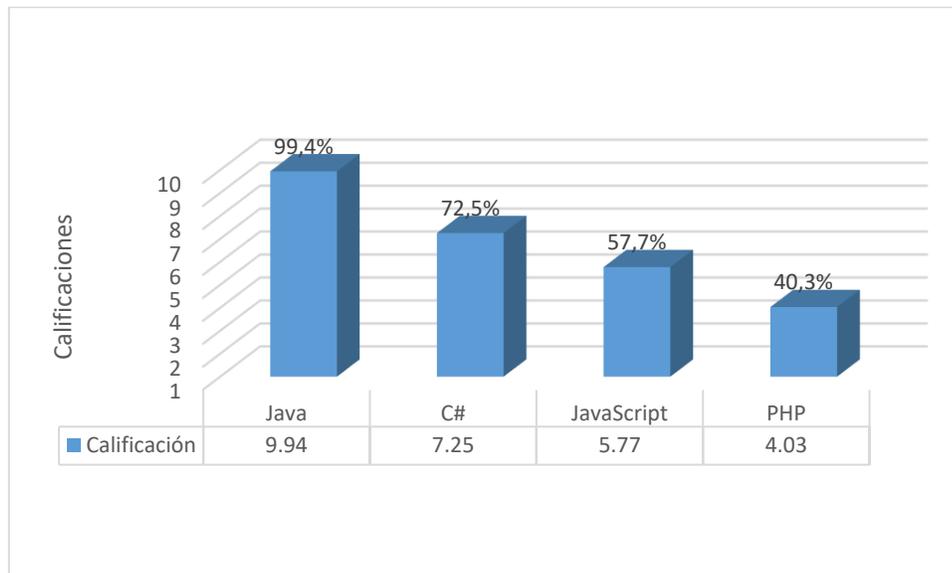


Figura. 42 Comparativa de Resultados entre lenguajes de programación.

Fuente: Propia

Como se puede observar en la *Figura. 42*, en base a las evaluaciones realizadas, se concluye que, el lenguaje de programación más eficiente para generar códigos QR es Java, esto basado en su calificación de 9.5/10. Por otro lado, la evaluación realizada a los lenguajes de programación dio como resultado lo siguiente: C# arrojó como resultado una calificación regular, JavaScript una calificación media, y PHP una calificación baja que difiere significativamente del mejor promedio.

3.3 Tabla resumen del análisis descriptivo

En la siguiente tabla se puede apreciar el análisis descriptivo realizado, para ellos se consideraron Factores de evaluación que denoten el comportamiento, rendimiento y eficiencia de cada lenguaje. También se observa los servidores, librerías y las versiones que se utilizaron para la implementación.

TABLA 20 Tabla resumen del análisis descriptivo

Tabla Resumen de Generadores QR					
Factores	Java (JSF)	C# (.NET)	JavaScript (Angular)	PHP	
Versión	2,2	2,1	7	7	
Servidor Utilizado	WidFly 10,0	ISS Expres	Angular CLI	Xamp 2,3,2	
Librerías	Primefaces v7.0 Barcode4j-light-2.1 core-3.3.3 javase-3.3.0 qrngen-1.4	MessagingToolkit. QRCode v1.3.	Angularx-qrcode	Phpqrcode versión 2010100721_1.14	
Tiempo de Respuesta del código QR	1	0,031 segundos	0,030 segundos	0,614 segundos	0,067 segundos
	1000	0,442 segundos	17,300 segundos	10,4671 segundos	30,763 segundos
Tiempo de Espera del código QR	1	0,002 segundos	0,005 segundos	0,558 segundos	0,040 segundos
	1000	0,380 segundos	12,309 segundos	9,930 segundos	29,8234 segundos
Rendimiento del código QR	1	35 pt por segundo	38 pt por segundo	11 pt por segundo	200 pt por segundo
	1000	12 pt por segundo	2 pt por segundo	1 pt por segundo	3 pt por segundo
Utilización de recursos	1	0,011 segundos	0,003 segundos	0,012 segundos	0,05 segundos
	1000	3,34 segundos	7,98 segundos	1,490 segundos	3,36 segundos
Utilización de la memoria	1	48,134 MB	20,78 MB	18,116 MB	2,44 MB
	1000	257,686 MB	153,64 MB	16,470 MB	121,75 MB

Fuente: Propia

3.4 Análisis de impacto

Este trabajo promoverá la innovación en todos los sectores de la sociedad del Ecuador debido a el nivel de innovación y amplia proyección que tiene, las tecnologías analizadas son relativamente nuevas y son aplicables a múltiples áreas de la ciencia e industria, lo que potenciará su uso, ya que presenta una nueva opción para automatización de procesos. En coherencia con los Objetivos de Desarrollo Sostenible y el plan Nacional de Desarrollo Toda una vida, este trabajo pretende que además del aporte que implica su realización, se abran nuevas líneas de investigación sobre el tema y se consiga nuevos conocimientos y

consecuentemente nuevas oportunidades de aprendizaje y aportes tecnológicos para la sociedad.

Con este trabajo se deja una tabla resumen del análisis descriptivo sobre los complementos (librerías) de códigos QR en los diferentes lenguajes de programación, la cual ayudara a los desarrolladores de software a elegir la mejor opción para realizar aplicaciones web que generen códigos QR con información específica.

CONCLUSIONES

- Al finalizar la investigación e implementación de los diferentes complementos de programación en los lenguajes propuestos, se establece que, el lenguaje de programación C# presenta dificultades, debido a que actualmente para este lenguaje no existe suficiente información en internet. Por otro lado, JavaScript, al ser un lenguaje de código abierto cuenta con gran variedad de información y actualizaciones de sus complementos, lo que facilita la implementación, sin embargo, esto puede generar conflicto sobre que versión de librería se debe utilizar.
- Se determinó que, el lenguaje de programación Java es la mejor opción para generar códigos QR en una aplicación Web, esto debido a que obtuvo una calificación alta que se considera “**satisfactoria**”, para obtener este resultado se aplicó las métricas de la característica 2 de la ISO/IEC 25010.
- Los resultados del estudio descriptivo sobre los lenguajes de programación propuestos establecieron puntos relevantes y precisos que sirven como base teórica fundamental para elegir las tecnologías más eficientes al implementar códigos QR en aplicaciones Web.
- Para generar los códigos QR en los demos desarrollados, se utilizó la base de datos de los estudiantes de Ingeniería en Sistemas Computacionales como material de apoyo, además de las herramientas para trabajar con esta información como son Excel, Postgres y J-Meter.

RECOMENDACIONES

- Antes de implementar algún tipo de generador de códigos QR en una aplicación Web, se debe investigar las versiones existentes en cada uno de ellos, además utilizar complementos estables.
- Investigar más sobre las funcionalidades de los códigos QR, actualmente en el país se utilizan los códigos QR para fines publicitarios, sin embargo, proveen más funcionalidades, entre ellas su uso en la automatización de procesos.
- Socializar a los estudiantes de la carrera las funcionalidades, beneficios y el impacto que están teniendo los códigos QR en la actualidad, con el fin de que se generen proyectos en los cuales se utilice esta tecnología.

BIBLIOGRAFÍA Y LINKOGRAFÍA

- Alicante, U. de. (2016). Universidad de Alicante. Recuperado de <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>
- Charoensiriwath, C., Surasvadi, N., Pongnumkul, S., & Pholprasit, T. (2015). Applying QR Code and Mobile Application to Improve Service Process in Thai Hospital. *International Joint Conference on Computer Science and Software Engineering (JCSSE)*.
- Domínguez, P. (2017). OPENCLASSROOMS. Recuperado de <https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>
- EcuRed. (s/f). Lenguaje de Programación. Recuperado de https://www.ecured.cu/Lenguaje_de_Programación
- Enríquez, L. J. (2016). *Estudio comparativo de herramientas orientadas al desarrollo de aplicaciones móviles, con el aplicativo: sistema de administración de stock de productos de consumo masivo*. Universidad Tecnica del Norte.
- Foundation, L. (2018). node js. Recuperado de <https://nodejs.org/es/about/>
- Google. (2019). ANGULAR. Recuperado de <https://angular.io/docs>
- Grados, J. (2017). ¿Qué es JavaScript? Recuperado de <https://devcode.la/blog/que-es-javascript/>
- Heflo. (2017). Que es la automatizacion de proceso? Recuperado de <https://www.heflo.com/es/blog/automatizacion-procesos/que-es-la-automatizacion-de-procesos/>
- Hellemons, J. (2012). nuget. Recuperado de <https://www.nuget.org/packages/MessagingToolkit.QRCode/>
- Holzner, S. (2009). *PHP*. McGRAW-HILL.

- Huidrobo, J. M. (2009). *Qué es Código QR*.
- Hussain, A., & Mkpojiogu, E. O. . (2015). AN APPLICATION OF THE ISO/IEC 25010 STANDARD IN THE QUALITY-IN-USE ASSESSMENT OF AN ONLINE HEALTH AWARENESS SYSTEM. *Jurnal Teknologi*, 5.
- ISO. (2018). ISO 25000. Recuperado de <https://www.iso25000.com/index.php/normas-iso-25000/iso-25010>
- Jacob, A. (2018). angularx-qrcode. Recuperado de <https://www.npmjs.com/package/angularx-qrcode>
- Karn, P. (2006). PHP QR Code. Recuperado de <http://phpqrcode.sourceforge.net/>
- Kim, J. H., Kim, M., Yang, T., Kim, I., Seo, J., & Kang, S. (2018). Compressed QR code-based mobile voice guidance service for the visually disabled. *International Conference on Advanced Communications Technology*, 3.
- McGraw-Hill. (2010). *Fundamentos de C# 3.0*. ProQuest Ebook Central. Recuperado de <http://ebookcentral.proquest.com/lib/utnortesp/detail.action?docID=3196018>
- Microsoft. (2018). Microsoft. Recuperado de <https://www.asp.net/>
- MINTEL. (2017). Ministerio de telecomunicaciones y de la sociedad de la información. Recuperado de <https://www.telecomunicaciones.gob.ec/crecimiento-economico-ecuatoriano-de-la-mano-con-la-tecnologia/>
- MvnRepository. (2015). MVNREPOSITORY. Recuperado de <https://mvnrepository.com/artifact/net.sf.barcode4j/barcode4j-light/2.1>
- MvnRepository. (2018). MVNREPOSITORY. Recuperado de <https://mvnrepository.com/artifact/com.google.zxing/javase/3.3.3>
- Npgsql. (2018). Npgsql. Recuperado de <http://www.npgsql.org/>
- Nuget. (2018). Nuget. Recuperado de <https://www.nuget.org/>
- Perez-Sanagustín, M., Parra, D., Verdugo, R., Gonzalo García-Galleguillos, & Nussbaum, M.

- (2016). Using QR codes to increase user engagement in museum-like spaces. *www.elsevier.com/locate/comphumbeh*, 13.
- Pineda Bohórquez, F. J. (2014). *Diseño e implementación de códigos QR en los equipos de la Empresa Automation Colombia "Ingeniería y Equipos Industriales de Colombia"*. UNIVERSIDAD PEDAGÓGICA Y TECNOLÓGICA DE COLOMBIA.
- PrimeTek. (2019). Primefaces. Recuperado de <https://www.primefaces.org/>
- QuestionPro. (2017). QuestionPro. Recuperado de <https://www.questionpro.com/blog/es/analisis-descriptivo/>
- Schildt, H. (2007). *Fundamentos de Java (3a. ed.)*. McGraw-Hill Interamericana.
- Schildt, H. (2009). *Java: manual de referencia (7a. ed.)*. McGraw-Hill Interamericana. Recuperado de <http://ebookcentral.proquest.com/lib/utnortesp/detail.action?docID=3191840>
- SEMPLADES. (2017). Plan Nacional de Desarrollo 2017-2021.Toda una Vida. *SEMPLADES*, 148.
- StrongLoop. (s/f). Express. Recuperado de <https://expressjs.com/es/>
- Unidas, N., & CEPAL. (2016). Agenda 2030 y los objetivos de Desarrollo Sostenible Una oportunidad para América Latina y el Caribe, 50.
- UnitagQR. (2011). UnitagQR. Recuperado de <https://www.unitag.io/es/qrcode/what-is-a-qrcode>
- UTEL. (2017). Historia del lenguaje Javascript. Recuperado de <https://www.utel.edu.mx/blog/menu-profesional/historia-del-lenguaje-javascript/>
- Vaswani, V. (2010). *Fundamentos de PHP*. (M.-H. Interamericana, Ed.). ProQuest Ebook Central. Recuperado de <https://ebookcentral.proquest.com/lib/bibliocauladechsp/reader.action?docID=3191866&query=PHP>
- Violeta_1879354. (2008). Características de C#. Recuperado de

<https://es.scribd.com/doc/7411856/Caracteristicas-de-C>

Web, L. (2015). Introducción a JavaScript. Recuperado de <https://librosweb.es/libro/javascript/capitulo-1/breve-historia.html>

Wikipedia. (2019). Wikipedia. Recuperado de <https://es.wikipedia.org/wiki/Implementación>

Zhou, Y., Hu, J., Yuan, S., Zhang, L., & Huo, D. (2018). Method of multiple-image hiding in QR code based on compressed sensing and orthogonal modulation. *www.elsevier.de/ijleo*, 10.

Alicante, U. de. (2016). Universidad de Alicante. Recuperado de <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>

Charoensiriwath, C., Surasvadi, N., Pongnumkul, S., & Pholprasit, T. (2015). Applying QR Code and Mobile Application to Improve Service Process in Thai Hospital. *International Joint Conference on Computer Science and Software Engineering (JCSSE)*.

Domínguez, P. (2017). OPENCLASSROOMS. Recuperado de <https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>

EcuRed. (s/f). Lenguaje de Programación. Recuperado de https://www.ecured.cu/Lenguaje_de_Programación

Enríquez, L. J. (2016). *Estudio comparativo de herramientas orientadas al desarrollo de aplicaciones móviles, con el aplicativo: sistema de administración de stock de productos de consumo masivo*. Universidad Tecnica del Norte.

Foundation, L. (2018). node js. Recuperado de <https://nodejs.org/es/about/>

Google. (2019). ANGULAR. Recuperado de <https://angular.io/docs>

Grados, J. (2017). ¿Qué es JavaScript? Recuperado de <https://devcode.la/blog/que-es-javascript/>

- Heflo. (2017). Que es la automatizacion de proceso? Recuperado de <https://www.heflo.com/es/blog/automatizacion-procesos/que-es-la-automatizacion-de-procesos/>
- Hellemons, J. (2012). nuget. Recuperado de <https://www.nuget.org/packages/MessagingToolkit.QRCode/>
- Holzner, S. (2009). *PHP*. McGRAW-HILL.
- Huidrobo, J. M. (2009). *Qué es Código QR*.
- Hussain, A., & Mkpojiogu, E. O. . (2015). AN APPLICATION OF THE ISO/IEC 25010 STANDARD IN THE QUALITY-IN-USE ASSESSMENT OF AN ONLINE HEALTH AWARENESS SYSTEM. *Jurnal Teknologi*, 5.
- ISO. (2018). ISO 25000. Recuperado de <https://www.iso25000.com/index.php/normas-iso-25000/iso-25010>
- Jacob, A. (2018). angularx-qr-code. Recuperado de <https://www.npmjs.com/package/angularx-qr-code>
- Karn, P. (2006). PHP QR Code. Recuperado de <http://phpqr-code.sourceforge.net/>
- Kim, J. H., Kim, M., Yang, T., Kim, I., Seo, J., & Kang, S. (2018). Compressed QR code-based mobile voice guidance service for the visually disabled. *International Conference on Advanced Communications Technology*, 3.
- McGraw-Hill. (2010). *Fundamentos de C# 3.0*. ProQuest Ebook Central. Recuperado de <http://ebookcentral.proquest.com/lib/utnortesp/detail.action?docID=3196018>
- Microsoft. (2018). Microsoft. Recuperado de <https://www.asp.net/>
- MINTEL. (2017). Ministerio de telecomunicaciones y de la sociedad de la información. Recuperado de <https://www.telecomunicaciones.gob.ec/crecimiento-economico-ecuadoriano-de-la-mano-con-la-tecnologia/>
- MvnRepository. (2015). MVNREPOSITORY. Recuperado de <https://mvnrepository.com/artifact/net.sf.barcode4j/barcode4j-light/2.1>

- MvnRepository. (2018). MVNREPOSITORY. Recuperado de <https://mvnrepository.com/artifact/com.google.zxing/javase/3.3.3>
- Npgsql. (2018). Npgsql. Recuperado de <http://www.npgsql.org/>
- Nuget. (2018). Nuget. Recuperado de <https://www.nuget.org/>
- Perez-Sanagustín, M., Parra, D., Verdugo, R., Gonzalo García-Galleguillos, & Nussbaum, M. (2016). Using QR codes to increase user engagement in museum-like spaces. *www.elsevier.com/locate/comphumbeh*, 13.
- Pineda Bohórquez, F. J. (2014). *Diseño e implementación de códigos QR en los equipos de la Empresa Automation Colombia "Ingeniería y Equipos Industriales de Colombia"*. UNIVERSIDAD PEDAGÓGICA Y TECNOLÓGICA DE COLOMBIA.
- PrimeTek. (2019). Primefaces. Recuperado de <https://www.primefaces.org/>
- QuestionPro. (2017). QuestionPro. Recuperado de <https://www.questionpro.com/blog/es/analisis-descriptivo/>
- Schildt, H. (2007). *Fundamentos de Java (3a. ed.)*. McGraw-Hill Interamericana.
- Schildt, H. (2009). *Java: manual de referencia (7a. ed.)*. McGraw-Hill Interamericana. Recuperado de <http://ebookcentral.proquest.com/lib/utnortesp/detail.action?docID=3191840>
- SEMPLADES. (2017). Plan Nacional de Desarrollo 2017-2021.Toda una Vida. *SEMPLADES*, 148.
- StrongLoop. (s/f). Express. Recuperado de <https://expressjs.com/es/>
- Unidas, N., & CEPAL. (2016). Agenda 2030 y los objetivos de Desarrollo Sostenible Una oportunidad para América Latina y el Caribe, 50.
- UnitagQR. (2011). UnitagQR. Recuperado de <https://www.unitag.io/es/qrcode/what-is-a-qrcode>
- UTEL. (2017). Historia del lenguaje Javascript. Recuperado de

<https://www.utel.edu.mx/blog/menu-profesional/historia-del-lenguaje-javascript/>

Vaswani, V. (2010). *Fundamentos de PHP*. (M.-H. Interamericana, Ed.). ProQuest Ebook Central. Recuperado de <https://ebookcentral.proquest.com/lib/bibliocauladechsp/reader.action?docID=3191866&query=PHP>

Violeta_1879354. (2008). Características de C#. Recuperado de <https://es.scribd.com/doc/7411856/Caracteristicas-de-C>

Web, L. (2015). Introducción a JavaScript. Recuperado de <https://librosweb.es/libro/javascript/capitulo-1/breve-historia.html>

Wikipedia. (2019). Wikipedia. Recuperado de <https://es.wikipedia.org/wiki/Implementación>

Zhou, Y., Hu, J., Yuan, S., Zhang, L., & Huo, D. (2018). Method of multiple-image hiding in QR code based on compressed sensing and orthogonal modulation. *www.e/sevier.de/ijleo*, 10.