CAPITULO II

MARCO TEÓRICO

2.1. Competencias digitales

2.1.1. Definición e importancia

2.1.2. Características

2.1.3. Estructura

2.2. Framework de competencias digitales: DigComp 2.1

2.2.1. Definición e importancia

2.2.2 Características

2.2.3. Estructura

2.3. Otros frameworks

2.4. Aplicativo Web

El Aplicativo Web comprende las diferentes metodologías de desarrollo, los lenguajes de programación y las bases de datos que se utilizarán para desarrollar el sistema. Luego se realizará un análisis comparativo de sus características, beneficios y limitaciones, con la finalidad de obtener los elementos de juicio necesarios, que guíen la selección de las mejores herramientas para el presente proyecto y poder desechar las demás opciones.

2.4.1. Metodología de desarrollo

La Literatura revisada nos ofrece una gran variedad de metodologías, pero en función de las ventajas y limitaciones de cada una de ellas; así como las características y necesidades del proyecto a implementar se han seleccionado tres para analizarlas y de entre ellas elegir la mejor.

1. **Scrum.**

Es una metodología ágil para el desarrollo de software, porque necesita de poca documentación frente a métodos tradicionales; está destinada al desarrollo y elaboración de código fuente. Es una metodología flexible, ya que permite la planificación diaria del trabajo; además de facilitar los cambios repentinos o inesperados, adaptándose a ellos sin experimentar el proyecto ningún impacto general.

Para Urteaga (2015) es una metodología amigable, porque se enfoca primero en el ser humano y luego en los procesos; establece una amplia comunicación y colaboración entre el cliente y el equipo de desarrollo, permite al cliente obtener resultados funcionales al finalizar cada sprint, que suele tener una duración de 1 a 4 semanas, permitiendo comenzar a aplicar la funcionalidad del proyecto a término de cada sprint y no necesita esperar la finalización completa del proyecto para su implementación.

Según Behit (2012) la Metodología Scrum necesita de los siguientes roles y funciones:

* **Product owner,** que es el miembro encargado en comunicar la visión del producto al equipo de desarrollo, representa al dueño del sistema, teniendo así un mayor peso o mayor autoridad dentro del proyecto.
* **Scrum master**, es el que se encarga de evitar cualquier impedimento en el equipo de desarrollo para poder culminar exitosamente un sistema; por lo tanto, es la persona más creativa y productiva dentro del proyecto, sirve de enlace entre el product owner y el equipo de desarrollo.
* **Equipo de desarrollo**, es el responsable de codificar el trabajo; está conformado por desarrolladores, analistas, testers, arquitectos y diseñadores del sistema, quienes son responsables de determinar cómo realizar cada uno de los sprints, permitiendo así enfocarse en el sprint individual del grupo de desarrollo y consiguiendo una autonomía entre cada equipo (pp.30-39).

1. **Cascada.**

Es el modelo de proceso más conocido durante el desarrollo de software, ha sido utilizado por mucho tiempo. Maneja una secuencia lineal de desarrollo, orientada por las diferentes fases que lo componen, que pueden ser entre 5 y 7 fases, las mismas se ejecutan una vez durante la duración del proyecto.

Los entregables, que en este modelo se denominan hitos, corresponden a grandes partes del proyecto; una vez elaborados, los testers realizan las pruebas necesarias; en caso de existir errores, la corrección de los mismos toma más tiempo y esfuerzo de lo normal, ya que el hito abarca un gran tamaño.

Lo positivo es que siempre existirá una buena documentación del desarrollo, los costes y carga de trabajo para el equipo se definen al iniciar el proyecto. Entre las desventajas están: El usuario o dueño del proyecto no podrá revisarlo hasta que haya finalizado el desarrollo y en ocasiones, en el proceso, no se suelen detectar de manera temprana los fallos. Los cambios tras la revisión del cliente no resulta fácil implementarlos (Cervantes, 2012).

1. **Extreme Programing XP**

Es una metodología ágil de desarrollo de aplicaciones; para su implementación se propone trabajar al mismo tiempo, en equipos de 2 personas en un solo computador; esto supone un sustancial incremento de costos en el desarrollo, y una reducción significativa de la eficiencia en cantidad de personas y tiempo de desarrollo. Sin embargo, como beneficio encontramos, que a pesar del incremento del costo, disponemos de 2 personas brindando ideas significativas para la evolución y mejoramiento del proyecto (Meléndez, Gaitán y Pérez, 2016).

Según Behit (2012) con metodología extreme programing XP el ritmo de desarrollo es parejo y mantiene siempre ciertas prácticas a seguir, tales como:

* Buena comunicación entre el equipo de desarrollo y el cliente.
* Entregas pequeñas en versiones completas.
* Mantener un diseño simple pero funcional.
* Pruebas para constatar que todo lo desarrollado cumpla con las funcionalidades y requerimientos del cliente.
* Se requerirá la presencia frecuente del cliente, para que el mismo teste la funcionalidad del sistema (pp. 70-71).

2.4.2. Lenguajes de programación

Existen diversos lenguajes a los que se podría recurrir para realizar trabajos de programación web, sin embrago, para este proyecto se ha seleccionado tres para analizarlas y luego elegir el más adecuado.

1. **PHP**

Desde la visión de Naveen, Geyavalli, Sujani y Rajesh (2019) se explica que el procesador de hipertexto, también conocido como PHP es un conjunto de scripts escritos en base a lenguajes más robustos como Perl y C. Principalmente se utiliza para el desarrollo de páginas web y aplicaciones web dinámicas, es considerado un lenguaje de programación fácil y sencillo de entender y aprender, por lo cual es recomendado para desarrolladores inexpertos o nuevos. Este lenguaje de programación realiza el procesamiento de código por el lado del servidor, lo cual evita la interpretación por parte del navegador.

Para Crawford y Hassain (2017) entre las principales ventajas del PHP están:

* Es un lenguaje con amplio soporte de diferentes bases de datos y una eficiente integración en múltiples protocolos de internet, al igual que por ser un lenguaje no interpretativo por parte del navegador, este no necesita una declaración constante de variables.
* Además, como es uno de los lenguajes más utilizados, mantiene una gran comunidad activa de desarrolladores para la resolución de dudas y mejoramiento de versiones.

Es importante tener presente que PHP incorpora HTML, por lo que provoca una mayor carga en el servidor y una de las mayores desventajas que tiene es la escritura débil y la falta de variantes u opciones dentro de su gestionamiento de errores.

Otra desventaja en varios desarrollos basados en PHP es que la falta actualización en las versiones recientes pone en peligro la seguridad, estabilidad y usabilidad de la página web, a pesar de ser uno de los lenguajes más utilizados en la actualidad.

1. **Python con Django**

Es un lenguaje de programación de alto nivel, que se encuentra basado en la reutilización de código, lo interesante es que emplea una sintaxis clara y fácil de comprender y aprender. Este leguaje es fácil de escribir ya que la separación entre líneas no mantiene una separación mediante caracteres, sino la utilización de sangrías para la delimitación entre líneas (Lei, Ma & Than, 2014).

Además, Naveen, Geyavalli, Sujani, Rajesh (2019) agregan, que actualmente muchas aplicaciones, que necesitan de una amplia seguridad y manejo de datos, han migrado a Python como lenguaje de codificación, entre las razones están su consistencia, código fuente limpio, la amplia funcionalidad de su biblioteca y una comunidad activa.

1. Ruby

Es considerado como un lenguaje interpretado, con muchas similitudes a otros lenguajes de programación, como Python o Perl; adopta su sintaxis del idioma inglés. Es un lenguaje de programación orientado a objetos, mantiene una diferenciación entre caracteres y escritura dinámica; la reflexión sobre objetos y listas está al entendimiento del desarrollador más que a la computadora.

Su principal característica es que dentro del desarrollo todo es entendido como un objeto, sean clases o valores, esto significa que no tiene una diferenciación entre tipos de datos. Es un lenguaje que permite la generación de métodos, manipulación de herencias y la modificación de parámetros del lenguaje de programación para la conveniencia del desarrollador y al igual que Python mantiene una amplia biblioteca de paquetes.

Para Cumba y Barreno (2012) la principal desventaja radica en la deficiencia de detección de errores tipográficos dentro del código, lo que causa errores inesperados, los mismos que afectan significativamente el tiempo de ejecución del sistema, al igual que causa una sobrecarga significativa de operadores.

2.4.3. Base de datos

Se conocen variedad de bases de datos, de las que se han seleccionado dos para compararlas y poder decidir con mayor fundamento.

1. MYSQL

Es un sistema gestor de bases de datos, que entra en las categorías de programas de código abierto. Se encuentra escrito en base a lenguajes de programación como C y C++, ha sido probado en un amplio rango de configuraciones diferentes a lo largo del tiempo, igualmente, aunque fue escrito para lenguajes Linux, permite la instalación y configuración en Windows brindando amplio soporte para todos los sistemas operativos actuales.

Siguiendo a Cobo, Gómez, Pérez y Rocha (2005) se pude decir que MySQL es un sistema multicapa, que mantiene módulos separados entre sí, brindando una amplia independencia entre ellos, lo que ayuda significativamente para la estabilidad de cada módulo. Entre los aspectos positivos conviene destacar: La velocidad y robustez que tiene; la posibilidad de almacenar la información creando un directorio por cada celda ingresada, permitiendo así realizar respaldos de información de manera muy sencilla.

1. POSGRESQL

Es un gestor de bases de datos relacional, que se encuentra principalmente orientado a objetos, es de código abierto y está mantenido por una significativa comunidad de desarrolladores, colaboradores, etc. Presenta alta concurrencia, variedad de tipos de datos de forma nativa.

Una desventaja es, que al ser una herramienta relativamente nueva resulta desconocida para el mercado desarrollador. Además, este gestor al estar orientado principalmente a objetos, las tablas y tuplas son objetos. Al igual que MySQL mantiene una arquitectura cliente – servidor (Cambi y Zúñiga, 2006).

2.4.4. Comparativas

La tabla siguiente tiene la finalidad de analizar y comparar de manera sintética las principales características, beneficios y limitaciones de los componentes de una programación web, con el único objetivo de elegir las mejores herramientas para el proyecto a desarrollar.

|  |  |  |  |
| --- | --- | --- | --- |
| 1. **Metodología de desarrollo** | | | |
| **Características** | 1. **SCRUM** | 1. **Cascada** | 1. **Extreme Programing XP** |
| * Entregables de corto periodo de tiempo. * Reuniones diarias para estudiar los avances y cambios. * Equipos independientes para análisis y cumplimiento de objetivos. * Planteamiento de tiempos máximos de ejecución de objetivos. * Se obtienen resultados de forma casi inmediata al tener corto tiempo de entrega de los sprints. * Fomenta una mayor funcionalidad de software sobre excesiva documentación. * Promueve reuniones periódicas con el cliente a fin de notar cambios o mejoras a tiempo. | * Es el método más utilizado. * Si se cambia el orden de las fases, la calidad del producto disminuye. * Mantiene una estricta sucesión de fases para el desarrollo. * Cada fase realiza una documentación para iniciar su fase sucesora. * Necesita contar con requerimientos claros, completos y bien establecidos. * Requiere personal experimentado. * Los errores de documentación y levantamiento de requerimientos se detectan de forma tardía. | * Metodología basada en pruebas y errores. * Se encuentra orientada principalmente al desarrollador y cliente del software. * Supone iniciar por las tareas más sencillas que puedan funcionar. * Fomenta la comunicación entre todas las partes involucradas en el proyecto. * Mantiene estrictas normas de codificación. * Se lo desarrolla en parejas de programadores. * Primero se planifican las pruebas a realizar para validar el proyecto. |
| **Beneficios** | * Existe un producto funcional tras cada sprint. * Mayor eficiencia en el desarrollo. * Reuniones diarias del equipo para conocer el estado del proyecto. * Exhaustiva revisión en los sprints. * Alta capacidad de cambios en los entregables. * Productividad y calidad. * Reunión frecuente con el cliente. * Entregables en 1 a 4 semanas. | * Abundante documentación del proyecto. * Todo se encuentra organizado y bien definido sin mayores cambios. * La planificación de desarrollo es sencilla. * La calidad de proyecto elaborado con este método es elevada. * Costes y carga de trabajo se determinan al iniciar el proyecto. * Las fases se estructuran cronológicamente de manera sencilla. * Facilita la gestión de desarrollo. | * Fomenta el trabajo en grupo. * Metodología basada en prueba y error. * Reducción de costes de los cambios en entregables. * Alta capacidad de cambios en el sistema. * Permite cambios en requerimientos. * Posibilita continuos diálogos entre todos los actores del proyecto. |
| **Limitaciones** | * Menor documentación final del producto terminado. * No puede aplicarse en todos los proyectos. | * Poca posibilidad de cambios una vez entregados. * Disminuye la cantidad de reuniones para saber el estado del proyecto. * Anticuado modelo de desarrollo. * Mala solución para proyectos complicados. * Demora entre pasos de cada fase. | * Necesita mayor cantidad de recursos en personal. * Se requiere un equipo con alta formación en cada uno de los temas. * Mayor costo de desarrollo. |
| **Opinión** | **SCRUM**  En la actualidad han incrementado la cantidad de proyectos que se realizan con este método.  Dependiendo de la amplitud del sistema la adaptación del método suele ser mayor, y para proyectos de mediana o pequeña dimensión ayudan significativamente los sprint.  Otra ventaja es que permite la revisión y socialización constante del desarrollo con el cliente.  Por lo tanto, es el mejor método de desarrollo para el proyecto actual. | **Cascada**  Actualmente sigue siendo el método más utilizado, aunque de manera obsoleta ya que necesita de mucha documentación en cada fase; y, para proyectos pequeños o medianos no tiene adaptación a cambios por el cliente.  Además, se requiere culminar el proyecto completo para que comience a funcionar.  Sirve principalmente para proyectos a gran escala y no se adapta a las necesidades actuales. | **Extreme Programing XP**  Es una metodología de desarrollo ágil, al igual que SCRUM, se adapta a variedad de proyectos desarrollados en grupo.  Permite al igual que SCRUM cambios tempranos en las entregas y dispone de mucha comunicación con el cliente final; pero, la planificación es más larga que SCRUM.  Siendo esta la razón por la cual no se la ha seleccionado como metodología de desarrollo. |

|  |  |  |  |
| --- | --- | --- | --- |
| 1. **Lenguajes de programación** | | | |
| **Características** | 1. **PHP** | 1. **Python con Django** | 1. **Ruby** |
| * Permite la declaración de variables sin expresar el tipo al que pertenece. * Es un lenguaje sencillo y fácil de utilizar. * Mantiene una plataforma independiente libre de cualquier SO. * Amplia documentación para los desarrolladores y resolución de problemas. * Permite la programación orientada a objetos, aunque no es su función principal. * Mejoras constantes de rendimiento y seguridad en sus nuevas versiones. * Es un código que se ejecuta en el servidor, regresando así solo el HTML al navegador. | * Se puede utilizar junto al framework de Django, que actualmente es uno de los mejores frameworks de desarrollo web, debido a su seguridad, flexibilidad y amplia documentación. * Permite utilizar un API robusto para bases de datos. * Emplea el modelo MVT (modelo - vista – template) similar al MVC. * Toda la documentación necesaria se encuentra accesible mediante su aplicación administrativa. * Mantiene un administrador configurado por defecto. * Permite consultas sin la necesidad de utilizar SQL mediante su ORM. * Fomenta la reutilización de código. | * Está mayormente enfocado al desarrollo más que a la maquina en donde se ejecute. * Su sintaxis se encuentra escrita en lenguaje casi natural. * Todo dato es un objeto. * Se puede modificar el lenguaje. * Buen manejo de excepciones. * Es un lenguaje interpretativo. * Es un lenguaje de alto nivel y de software libre. * Emplea un paradigma POO, impidiendo que las variables puedan verse desde fuera de la clase. |
| **Beneficios** | * Lenguaje sencillo de aprender. * Multiplataforma. * No requiere definición de variables. * Puede ser combinado con HTML. * Hay abundante documentación. * Libre y gratuito. * Mantiene una comunidad muy amplia. * Maneja excepciones. | * Libre y de código abierto. * Lenguaje progresivo. * Multiplataforma. * Orientado a objetos. * Portable. * Amplia reusabilidad de funciones. * Aumento considerable de aplicaciones que están realizándose con este lenguaje. * No requiere una compilación para funcionar. * Librerías extendidas. | * Orientado a objetos. * Diferencia entre mayúsculas y minúsculas. * Maneja excepciones. * Desarrollo de bajo costo. * Multiplataforma. * Portable. * Software libre. |
| **Limitaciones** | * Es deficiente para aplicaciones grandes. * Todo el trabajo se realiza en el servidor. * El código fuente no puede ser ocultado. | * Lenguajes interpretativos suelen ser relativamente lentos. * Necesita una sangría bien estructurada para realizar los bloques de código. | * Es relativamente nuevo y posee menos documentación. * No es muy difundido o conocido por la comunidad web. |
| **Opinión** | **PHP**  Es uno de los lenguajes más utilizados, pero requiere un mayor esfuerzo por parte del servidor, es más pesado y no mantiene una reusabilidad avanzada.  La seguridad es otro tema importante al no permitir que cualquier usuario pueda ver el código fuente. | **Python**  Es un lenguaje de programación de alto nivel, el cual nos permite adaptarnos muy bien con Django.  Es un framework para desarrollo web que se encuentra actualmente en gran cantidad de proyectos, debido a la seguridad que nos ofrece tanto, por la separación en capas, como por los modelos de base de datos para evitar inyecciones SQL e intromisiones en su base de datos, lo convierten en un lenguaje con framework muy bueno. | **Ruby**  Es un lenguaje que aumenta su usabilidad en sistemas web; pero al ser muy sencillo no es lo suficientemente conocido por la comunidad de desarrollo. Posee un reducido soporte y poca documentación, aunque su uso crece paulatinamente. |

|  |  |  |
| --- | --- | --- |
| 1. **Bases de datos** | | |
| **Características** | 1. **MYSQL** | 1. **POSTGRESQL** |
| * Permite la selección de múltiples opciones de almacenamiento dependiendo de las necesidades de cada tabla. * Mantiene un amplio lenguaje SQL. * Dispone de soporte en todos los sistemas operativos. * Facilita la replicación de tablas. * Soporta altas cargas de almacenamiento. * Contiene un sistema de contraseñas seguro y verificación de tráfico para conectarse al servidor. * Mantiene una arquitectura flexible y fácil de entender. | * Presenta un sistema denominado MVCC para alta concurrencia. * Permite la consulta de información en la base de datos mientras ésta se encuentre en modo de recuperación. * Es una base de datos ACID. * Posee un amplio juego de caracteres internacionales. * Mantiene una encriptación SSL. * El lenguaje SQL utilizado permite cercanía o similitud al estándar ISO/IEC. |
| **Beneficios** | * No necesita de alta carga de requerimientos computacionales para su ejecución. * Instalación y configuración sencilla. * Bajo costos para una buena ejecución del programa. * Permite la encriptación de información. * Soporte en todos los sistemas operativos actuales. * Mantiene una licencia GPL. * Arquitectura cliente – servidor. | * Instalación ilimitada en todos nuestros equipos. * Gran capacidad de escalabilidad. * Está diseñado para alto volumen de trabajo. * Mantiene una fluida herramienta gráfica para la administración de bases de datos. * Permite mucha estabilidad al poder realizar consultas con las bases de datos en suspensión. * Gran capacidad de almacenamiento. |
| **Limitaciones** | * Muchas utilidades no presentan documentación. * Tiene limitaciones en el procesamiento de búsquedas. * No es intuitivo. * No maneja un lenguaje referencial. * Lento con grandes bases de datos. | * Es relativamente lento en ingresos y actualizaciones de información en bases de datos pequeñas. * No cuenta con soporte oficial en línea ni telefónico, solo cuenta con foros oficiales. * Necesita un nivel medio de conocimientos en lenguaje SQL. * Complejidad de comandos o sintaxis. * Más lento en inserciones y actualizaciones que MySQL. |
| **Opinión** | **MySQL**  Es el gestor de bases de datos más utilizado en la actualidad, mantiene una amplia capacidad de seguridad gracias a su gestión de usuarios; a pesar de que muchas funcionalidades no presentan documentación; en forma general, la documentación del gestor es amplia y suficiente para aprender completamente.  MySQL también posee amplio soporte de todos los sistemas operativos y mantiene bajos costes de ejecución.  Por todas estas razones se ha seleccionado MySQL como base de datos para el presente proyecto. | **PostgreSQL**  Es un gestor de bases de datos que permite amplia capacidad de concurrencia; pero, a pesar de su gran capacidad de usabilidad con las bases de datos en recuperación o suspensión, realiza un consumo de recursos mayor al de MySQL.  También es más lento en los procesos de inserción y de actualización de información y, al no encontrase diseñado para aplicaciones de pequeña o mediana escala crea una amplia desventaja con respecto a MySQL. |