La elaboración de la SLR, fue la base para la elección de la metodología, en los trabajos analizados se identificaron dos metodologías entre las que destacaron SCRUM y XP, sin embargo para el siguiente apartado se toma en consideración algunas metodologías ágiles con la finalidad de desarrollar una tabla comparativa que permita la elección de las más adaptativa al trabajo de titulación.

**Selección de Metodología**

En la década de 1990 el descontento con los métodos tradicionales aplicado en la ingeniería de software condujo a algunos desarrolladores de software a proponer nuevos “métodos ágiles”, los cuales permitieron que el equipo de desarrollo se enfocara en el software en lugar del diseño y la documentación. Los métodos ágiles se apoyan en el enfoque incremental para la especificación, el desarrollo y la entrega del software. Son más adecuados para el diseño de aplicaciones en que los requerimientos del sistema cambian rápidamente en el proceso de desarrollo, uno de los objetivos que se tiene con esta metodología es la entrega con prontitud el software operativo a los clientes, ellos a su vez propondrán requerimientos nuevos que serán incluidos en futuras iteraciones del sistema. Los métodos agiles se dirigen a simplificar el proceso burocrático al evitar trabajo con valor dudoso a largo plazo [1].

Algunas de las principales metodologías que se pueden adaptar al presente proyecto se detallan a continuación:

* Programación Extrema (XP)
* Scrum
* Crystal
* DSDM

Se tomará en consideración un concepto de cada una de las metodologías, sus características y ciclo de vida, permitiendo realizar una tabla comparativa para la elección de la metodología adecuada al final del apartado.

**Programación Extrema (XP)**

La metodología ágil XP está centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico [2].

Algunas de las principales características que nos brinda la metodología XP [3][4]:

* Se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad.
* Se aplica de manera dinámica durante el ciclo de vida del software.
* Es capaz de adaptarse a los cambios de requisitos.
* Los individuos e iteraciones son más importantes que los procesos y herramientas.
* Las personas es el principal factor de éxito que se tiene al llevar a cabo esta metodología.
* Da lugar a una programación sumamente organizada.
* Ocasiona eficiencias en el proceso de planificación y pruebas.
* Cuenta con una tasa de errores muy pequeña.
* Propicia la satisfacción del programador.
* Fomenta la comunicación entre los clientes y los desarrolladores.
* Facilita los cambios.
* Permite ahorrar mucho tiempo y dinero.
* Puede ser aplicada a cualquier lenguaje de programación.
* El cliente tiene el control sobre las prioridades.
* Se hacen pruebas continuas durante el proyecto.
* La XP es mejor utilizada en la implementación de nuevas tecnologías.

**Las fases de la metodología XP** [3]**:**

* Planeación
* Diseño
* Codificación
* Pruebas

**SCRUM**

Scrum es un marco de trabajo de procesos que ha sido usado para gestionar el desarrollo de productos complejos desde principios de los años 90. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos. Scrum muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo, de modo que podamos mejorar.

Scrum es un marco de trabajo por el cual las personas pueden acometer problemas complejos adaptativos, a la vez que entregar productos del máximo valor posible productiva y creativamente.

El marco de trabajo Scrum consiste en los Equipos Scrum, roles, eventos, artefactos y reglas asociadas. Cada componente dentro del marco de trabajo sirve a un propósito específico y es esencial para el éxito de Scrum y para su uso [5].

Algunas de las principales características que nos brinda la metodología[4]**:**

* El coste en términos de proceso y Management es mínimo, llevando a un resultado más rápido y barato.
* Permite realizar proyectos en los que la documentación de los requerimientos de negocios no está muy claros como para ser desarrolladas
* Se desarrolla rápidamente y testea. Cualquier error puede ser fácilmente rectificado.
* Los problemas se identifican por adelantado en las reuniones diarias y por lo tanto se pueden resolver rápidamente.
* Iterativo en su naturaleza, requiere continuo feedback del usuario
* Fácil de manejar los cambios debido a los sprints tan cortos y el feedback constante.
* Las reuniones diarias hacen posible medir la productividad individual. Esto lleva a la mejor en la productividad de cada uno de los miembros del equipo.

**Las fases de la metodología Scrum** [6]**.**

* Preparación del proyecto.
* Planificar un Sprint.
* Desarrollo de un Sprint.
* Diagrama detallado de las fases del Scrum.

**CRYSTAL**

Crystal Clear no aspira a ser la “mejor” metodología; aspira a ser "suficiente", de tal manera que el equipo lo amolde a sus necesidades y lo use, es una metodología que funciona con personas, descripta de la manera más simple como siguiente:

* El diseñador líder y otros dos a siete desarrolladores en una gran habitación o habitaciones contiguas, con radiadores de información, como pizarrones y rotafolios en la pared, teniendo acceso a usuarios claves, distracciones mantenidas al margen, entregando y corriendo código usable y probado cada mes o dos (a lo sumo tres), reflexionando periódicamente y ajustando su propio estilo de trabajo.

La familia de metodologías Crystal comparten con la XP una orientación humana, pero esta centralización en la gente se hace de una manera diferente. Alistair considera que las personas encuentran difícil seguir un proceso disciplinado, así que más que seguir la alta disciplinada que aun podría tener éxito, intercambiando conscientemente productividad por facilidad de la ejecución[7].

Crystal considera que las personas como dispositivos activos, alguna de las características**[[1]](#footnote-1)** principales de esta metodología son[8] :

* Cuando el número de personas aumenta, también aumenta la necesidad de coordinar.
* Cuando el potencial de daños se incrementa, la tolerancia a variaciones se ve afectada.
* La sensibilidad del tiempo en que se debe estar en el mercado varía: a veces este tiempo debe acortarse al máximo y se toleran defectos, otras se enfatizan la auditoria, confiabilidad, protección legal, entre otros.
* Las personas se comunican mejor cara a cara, con la pregunta y la respuesta en el mismo espacio de tiempo.
* Entrega frecuente. Consiste en entregar software a los clientes con frecuencia, no solamente compilar el código. (La frecuencia dependerá del proyecto, pero puede ser diaria, semanal o mensual).
* Comunicación osmótica. Todos juntos en el mismo cuarto.  
  Una variante especial es disponer en la sala de un experto diseñador senior y discutir respecto del tema que se trate.
* Mejora reﬂexiva,tomarse un pequeño tiempo (unas pocas horas cada o una vez al mes) para pensar bien qué se está haciendo, comparar notas, discutir.
* Seguridad personal, hablar con los compañeros cuando algo molesta dentro del grupo.
* Foco, saber lo que se está haciendo y tener la tranquilidad y el tiempo para hacerlo.
* Fácil acceso a usuarios expertos, tener alguna comunicación con expertos desarrolladores.

**Las fases de la metodología Crystal.**

* Puesta en escena.
* Revisiones.
* Monitoreo.
* Paralelismo y Flujo.
* Estrategia de diversas holística.
* Técnica de puesta a punto de la metodología.
* Puntos de vista del usuario.

**DSDM (**Dynamic Systems Development Method**)** [9]

DSDM inicialmente fue creado en 1994 gracias a la colaboración de un gran número de practicantes de proyectos. DSDM reconoce que los proyectos son limitados por el tiempo y los recursos, y los planes acordes a las necesidades de la empresa.

Es un método que provee un marco de trabajo para el desarrollo, ágil de software, busca desarrollar un sistema que reúna las necesidades de la empresa en tiempo, presupuesto y calidad.

DSDM también aborda muchas de las preocupaciones generales sobre desarrollo ágil. Específicamente, DSDM requiere fundamentos básicos para el proyecto que se acuerden en una etapa temprana. Esto permite a las empresas a comprender el alcance y las características fundamentales de la solución propuesta, y la forma en que se creará, antes de que comience el desarrollo.

Algunas de las principales características con las que cuenta esta metodología son:

* La participación del cliente es fundamental para el adecuado progreso del proyecto.
* El equipo del proyecto es responsable de toma de decisiones importantes sin esperar aprobación de niveles superiores.
* DSDM se basa en la entrega frecuentes del producto, asume que la entrega de única de un proyecto al finalizar no cumpliría las expectativas del cliente.
* Centra sus esfuerzos en funcionalidades críticas para alcanzar metas establecidas en el proyecto.
* Todos los cambios durante el desarrollo son reversibles.
* El objetivo de la metodología DSDM es realizar el desarrollo de forma rápida, evitando errores al momento de trabajar con el cliente.
* A través de la entrega continua de entregables se facilita el cambio en el proceso de desarrollo.
* Es flexible al momento de la reutilización de código en base a proyectos que han trabajado con anterioridad.

**Las fases de la metodología DSDM.**

* Estudio de viabilidad
* Estudio de la empresa
* Iteración del modelo funcional
* Diseño e iteración de la estructura
* Implementación.

**Comparativa de las metodologías (XP, Scrum, Crystal, DSDM)**

La realización de la tabla comparativa se basa en los trabajos propuestos en [10][11] [12], dichos estudios nos permitió obtener las características de forma individual de cada una de las metodologías, se extrajo información de cada una de las fuentes y se crea una tabla que resume las características principales de cada metodología.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Características | **XP** | **Scrum** | **Crystal** | **DSDM** |
| Vista de utilización | | | | |
| Respeto de los plazos de entrega | 0 | 1 | 1 | 1 |
| Respeto de los requisitos | 1 | 1 | 1 | 1 |
| Respeto de un nivel de calidad | 0 | 0 | 0 | 0 |
| Satisfacción del usuario final | 0 | 0 | 0 | 1 |
| Entorno turbulento | 1 | 1 | 0 | 1 |
| Aumento de la productividad | 1 | 0 | 0 | 0 |
| Capacidad de visión de agilidad | | | | |
| Iteraciones cortas | 1 | 1 | 0 | 0 |
| Colaborativo | 1 | 0 | 1 | 1 |
| Centrado en las personas | 1 | 0 | 1 | 1 |
| Política de refactorización | 1 | 0 | 0 | 1 |
| Política de pruebas | 1 | 1 | 0 | 1 |
| Integración de los cambios | 1 | 1 | 0 | 1 |
| Peso ligero | 1 | 1 | 0 | 0 |
| Requisitos funcionales pueden cambiar | 1 | 1 | 0 | 1 |
| El plan de trabajo puede cambiar | 1 | 0 | 0 | 0 |
| Los recursos humanos pueden cambia | 1 | 0 | 1 | 0 |
| Indicadores de cambio | 1 | 0 | 0 | 0 |
| Reactividad | 1 | 1 | 1 | 1 |
| Tamaño del proyecto pequeño | 1 | 1 | 1 | 0 |
| Complejidad del proyecto bajos | 1 | 0 | 0 | 0 |
| Riesgos del proyecto bajos | 1 | 0 | 0 | 0 |
| Tamaño del equipo pequeños | 1 | 1 | 1 | 0 |
| Altas interacción con los clientes | 1 | 0 | 0 | 1 |
| Altas interacciones de los miembros del equipo | 1 | 0 | 1 | 1 |
| Enfocado a proyectos de desarrollo | 1 | 0 | 1 | 0 |
| Genera documentación básica | 1 | 1 | 1 | 1 |
| Rapidez en comunicación entre miembros del equipo | 1 | 1 | 0 | 0 |
| Reuniones de seguimiento constantes | 1 | 1 | 0 | 0 |
| Construcción o Codificación | | | | |
| El equipo puede ser multidisciplinario | 1 | 1 | 1 | 1 |
| Utiliza una Técnica de desarrollo Guiado por Pruebas | 1 | 1 | 0 | 0 |
| Rapidez en comunicación entre miembros del equipo | 1 | 1 | 0 | 0 |
| Resumen | 90,32% | 54,84% | 38,71% | 48,39% |

Con la realización de la tabla “x” se pudo obtener las diferentes características de las metodologías ágiles, se adquirió un total de 31 campos de evaluación equivalente al 100%, la metodología XP obtuvo un total de 28 puntos equivalente al 90.32%, la metodología Scrum obtuvo un total de 17 puntos equivalente al 54,84%, la metodología Crystal obtuvo un total de 12 puntos equivalente al 38,71% y la metodología DSDM obtuvo un total de 15 puntos equivalente al 48,39%.

Una vez realizada la tabla comparativa de metodologías agiles, se pudo observar que la metodología XP, brinda un mayor número de características, ventajas y fortalezas que se reflejan a través de la tabla “x”, dichas características apoyan a la utilización de la metodología XP en la realización del presente trabajo de titulación.

# Bibliography

[1] I. Sommerville, *Software Engineering*. 2011.

[2] K. BECK, *Una explicación de la programación extrema:Aceptar el cambio.* MADRID, 2002.

[3] Bustamante Dayana and Rodríguez Jean, “Metodología Actual-Metodología XP,” 2014.

[4] L. Florez, M. Felipe, G. Tobon, U. Tecnologica, D. E. Pereira, and F. De Ingenierias, “FORMULACION DE CRITERIOS PARA LA SELECCION DE METODOLOGIAS DE DESARROLLO DE SOFTWARE.”

[5] L. G. Definitiva, “La Guía de Scrum.”

[6] Manuel Trigas Gallego, “Metodología Scrum.”

[7] A. Cockburn, *Agile software development*. 2002.

[8] A. O. Duarte and M. Rojas, “Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo,” 2008.

[9] Oya Maria Rosa and Torrealba Javier, “MÉTODO DE DESARROLLO DE SISTEMAS DINÁMICOS (DSDM),” Venezule, 2016.

[10] A. Iacovelli and C. Souveyet, “Framework for Agile Methods Classification,” 2008.

[11] G. Ahmad, Soomro Tariq, and Nawas Mohammad, “Agile Methodologies : Comparative Study and Future Direction,” no. July, 2014.

[12] Madruñero Edwin, “Implementación del estándar ISO/IEC 29110 en el proceso de desarrollo de softawre de la dirección de desarrollo tecnológico e informático de la Universidad Técnica del Norte,” p. 213, 2018.

1. https://folderit.net/itech/es/desarrollo-agil-de-software-crystal-clear-es/ [↑](#footnote-ref-1)