

metodologia xp

ADMINISTRACION DE SISTEMAS DE INFORMACIÓN



13 de abril de 2018

instituto tecnologico del putumayo

METODOLOGIA XP

ADRIANA MARICELA PEJENDINO MUTUMBAJOY

BRAYAN AUGUSTO PEREZ TIMANA

MAYRA JANETH LOPEZ FLOREZ

JUAN PABLO ROSERO CARLOSAMA

DOCENTE

ING. SAYRA OCORO

INSTITUTO TECNOLOGICO DEL PUTUMAYO

FACULTAD DE INGENIERIA Y CIENCIAS BASICAS

INGENIERIA DE SISTEMAS

ADMINISTRACION DE SISTEMAS DE INFORMACION

MOCOA-PUTUMAYO

2018

TABLA DE CONTENIDO

[PROGRAMACIÓN XP 2](#_Toc511341842)

[CARACTERÍSTICAS FUNDAMENTALES 2](#_Toc511341843)

[Pruebas unitarias continuas 2](#_Toc511341844)

[Programación en parejas 2](#_Toc511341845)

[Integración del equipo de programación con el cliente o usuario 3](#_Toc511341846)

[Corrección de todos los errores antes de añadir nueva funcionalidad 3](#_Toc511341847)

[Refactorización del código 3](#_Toc511341848)

[Propiedad del código compartida 3](#_Toc511341849)

[Simplicidad en el código 3](#_Toc511341850)

[ACTIVIDADES DE XP 3](#_Toc511341851)

[Codificar 3](#_Toc511341852)

[Hacer pruebas 4](#_Toc511341853)

[Escuchar 4](#_Toc511341854)

[Diseñar 4](#_Toc511341855)

[ROLES EN XP 4](#_Toc511341856)

[Programador 4](#_Toc511341857)

[Cliente 4](#_Toc511341858)

[Encargado de pruebas (Tester) 5](#_Toc511341859)

[Encargado de seguimiento (Tracker) 5](#_Toc511341860)

[Entrenador (Coach) 5](#_Toc511341861)

[Consultor 5](#_Toc511341862)

[Gestor (Gran jefe) 5](#_Toc511341863)

[CINCO VALORES FUNDAMENTALES DE XP 6](#_Toc511341864)

[COMUNICIACION 6](#_Toc511341865)

[SIMPLICIDAD 6](#_Toc511341866)

[RETROALIMENTACIÓN DEL SISTEMA 6](#_Toc511341867)

[Retroalimentación del cliente (usuario) 6](#_Toc511341868)

[Retroalimentación del equipo 6](#_Toc511341869)

[CORAJE 7](#_Toc511341870)

[RESPETO 7](#_Toc511341871)

[ARTEFACTOS DE XP 8](#_Toc511341872)

[Historias de Usuario 8](#_Toc511341873)

[Casos de prueba de aceptación 8](#_Toc511341874)

[Tarjetas de ingeniería 8](#_Toc511341875)

[Tarjetas CRC (Clase - Responsabilidad – Colaborador) 8](#_Toc511341876)

[REFERENCIAS 9](#_Toc511341877)

# 

# PROGRAMACIÓN XP

La programación extrema (XP) es una metodología de desarrollo ágil que tiene como principal objetivo aumentar la productividad a la hora de desarrollar un proyecto software. Da prioridad a los trabajos que dan un resultado directo y en los cuales se reduce la burocracia que pueda existir en el entorno de trabajo.

El objetivo principal de XP es entregar un software de calidad controlado por las necesidades del cliente.

# CARACTERÍSTICAS FUNDAMENTALES

 Desarrollo iterativo e incremental: pequeñas mejoras, unas tras otras.

Pruebas unitarias continuas: frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión. Se aconseja escribir el código de la prueba antes de la codificación.

Programación en parejas: se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código escrito de esta manera, el código es revisado y discutido mientras se escribe, es más importante que la posible pérdida de productividad inmediata.

Integración del equipo de programación con el cliente o usuario: Se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.

Corrección de todos los errores antes de añadir nueva funcionalidad: Hacer entregas frecuentes.

Refactorización del código: es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad, pero sin modificar su comportamiento. Las pruebas han de garantizar que en la refactorización no se ha introducido ningún fallo.

Propiedad del código compartida: en vez de dividir la responsabilidad en el desarrollo de cada módulo en grupos de trabajo distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.

Simplicidad en el código: es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrá añadir funcionalidad si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo.

# ACTIVIDADES DE XP

Codificar: Es necesario codificar y plasmar nuestras ideas a través del código. En programación, el código expresa la interpretación del problema, así podemos utilizar el código para comunicar, para hacer comunes las ideas, y por tanto para aprender y mejorar.

Hacer pruebas: Las características del software que no pueden ser demostradas mediante pruebas simplemente no existen. Las pruebas dan la oportunidad de saber si lo implementado es lo que en realidad se tenía en mente. Las pruebas nos indican que nuestro trabajo funciona, cuando no podemos pensar en ninguna prueba que pudiese originar un fallo en nuestro sistema, entonces habremos acabado por completo.

Escuchar: Si vamos a hacer pruebas tenemos que preguntar si lo obtenido es lo deseado, y tenemos que preguntar a quién necesita la información. Tenemos que escuchar a nuestros clientes cuáles son los problemas de su negocio, debemos de tener una escucha activa explicando lo que es fácil y difícil de obtener, y la realimentación entre ambos nos ayudan a todos a entender los problemas.

Diseñar: El diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Los diseños deben de ser sencillos, si alguna parte del sistema es de desarrollo complejo, lo apropiado es dividirla en varias. Si hay fallos en el diseño o malos diseños, estos deben de ser corregidos cuanto antes.

Resumiendo, en las actividades de XP tenemos que codificar porque sin código no hay programas, hacer pruebas porque sin pruebas no sabemos si hemos acabado de codificar, escuchar, porque si no escuchamos no sabemos que codificar ni probar, y diseñar para poder codificar, probar y escuchar indefinidamente.

# ROLES EN XP

Programador: Escribe las pruebas unitarias y produce el código del sistema. Debe existir una comunicación y coordinación adecuada entre los programadores y otros miembros del equipo.

Cliente: Escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio. El cliente es sólo uno dentro del proyecto, pero puede corresponder a un interlocutor que está representando a varias personas que se verán afectadas por el sistema.

Encargado de pruebas (Tester): Encargado de pruebas ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.

Encargado de seguimiento (Tracker): Encargado de seguimiento proporciona realimentación al equipo en el proceso XP. Su responsabilidad es verificar el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, comunicando los resultados para mejorar futuras estimaciones.

También realiza el seguimiento del progreso de cada iteración y evalúa si los objetivos son alcanzables con las restricciones de tiempo y recursos presentes. Determina cuándo es necesario realizar algún cambio para lograr los objetivos de cada iteración.

Entrenador (Coach): Es responsable del proceso global. Es necesario que conozca a fondo el proceso XP para proveer guías a los miembros del equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.

Consultor: Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto. Guía al equipo para resolver un problema específico.

Gestor (Gran jefe): Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.

# CINCO VALORES FUNDAMENTALES DE XP

COMUNICIACION: Establece: “Todos son parte del equipo y nos comunicamos cara a cara todos los días. Trabajamos juntos en todo, desde los requerimientos hasta la programación. En equipo crearemos la mejor solución al problema.” Con XP la comunicación se realiza por medio de transferencia de conocimientos en reuniones frecuentes cara a cara entre usuarios y desarrolladores, lo que le da a ambos una visión compartida del sistema.

SIMPLICIDAD: Implica que: “Desarrollaremos lo que sea solicitado y necesario, pero no más que eso. De esa forma, se maximiza el valor de la inversión realizada, dirigiremos nuestro objetivo a pasos simples y pequeños, mitigando las fallas a medida que ocurran. Crearemos algo de lo cual podamos sentirnos orgullos y que pueda mantenerse en el largo plazo a costos razonables.”

En XP se comienza desarrollando las soluciones más sencillas necesarias para solucionar los problemas (requerimientos) que se están viendo en ese momento, añadiendo funcionalidad extra más tarde, en la medida en que se obtiene más información de los requerimientos.

RETROALIMENTACIÓN DEL SISTEMA: Por medio de la ejecución de pruebas unitarias y de integración, los programadores reciben retroalimentación directa del estado del sistema.

Retroalimentación del cliente (usuario): Las pruebas de aceptación, son diseñadas conjuntamente por el cliente y los analistas de pruebas, obteniendo en conjunto retroalimentación del estado actual del sistema. Esta revisión puede hacerse cada 2 o 3 semanas, permitiendo así que el cliente sea quien guíe el desarrollo del software.

Retroalimentación del equipo: Cuando el cliente trae nuevos requerimientos, el equipo puede directamente proporcionar la estimación del tiempo que tomará implementarlos.

Bajo este esquema, las fallas de sistema se pueden comunicar fácilmente, pues existen pruebas unitarias que demuestran que el sistema fallará si es puesto en producción. Asimismo, un cliente puede probar el sistema periódicamente, contrastando el funcionamiento con sus requerimientos funcionales o “Historias de usuario”.

CORAJE: Establece: “Diremos la verdad en nuestros avances y estimados, no documentaremos excusas para el fracaso, pues planificamos para tener éxito. No tendremos miedo a nada pues sabemos que nadie trabaja solo. Nos adaptaremos a los cambios cuando sea que estos ocurran,   
algunas prácticas del coraje son:

* Diseñar y programar para hoy y no para mañana, evitando así hacer énfasis en el diseño en detrimento de todo lo demás.
* Refactorizar el código siempre que sea necesario (No tener reservas al respecto).
* Inspeccionar constantemente el código y modificarlo (refactorizar), de tal manera que futuros cambios se puedan implementar más fácilmente (desarrollar rápido para atender las necesidades de hoy, pero refactorizar después para facilitar el mantenimiento).
* Desechar componentes o piezas de código cuando sea necesario, sin preocuparse del tiempo invertido (y perdido) en su creación (Es mejor desechar algo que no es útil en lugar de tratar de repararlo).
* Ser persistente en la resolución de problemas.

RESPETO: “Todos en el equipo dan y reciben el respeto que merecen como integrantes del equipo y los aportes de cada integrante son valorados por todos. Todos contribuyen, así sea simplemente con entusiasmo.

* + Los desarrolladores respetan la experticia de los clientes y viceversa.
  + La Gerencia respeta el derecho del equipo de asumir responsabilidad y tener autoridad sobre su trabajo”.

Respeto es tanto por el trabajo de los demás como por el trabajo de uno mismo. Asimismo, los desarrolladores respetan su propio trabajo por medio de su compromiso con una alta calidad y buscando el mejor diseño para la solución por medio de la refactorización constante.

# ARTEFACTOS DE XP

Historias de Usuario: Son tarjetas físicas en las cuales se anota una descripción de una funcionalidad del sistema, en una oración, se le da un número y un título para ser identificada.

Casos de prueba de aceptación: Son tarjetas que se elaboran para realizar las pruebas de cada historia de usuario.

Tarjetas de ingeniería: Son tarjetas que se elaboran para ayudar y simplificar la programación de una historia de usuario.

Tarjetas CRC (Clase - Responsabilidad – Colaborador): Estas tarjetas se dividen en tres secciones que contienen la información del nombre de la clase, sus responsabilidades y sus colaboradores.

* Una clase es cualquier persona, cosa, evento, concepto, pantalla o reporte.
* Las responsabilidades de una clase son las cosas que conoce y las que realizan, sus atributos y métodos.
* Los colaboradores de una clase son las demás clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades.

# REFERENCIAS

pmoinformatica.com. (12 de 04 de 2018). Obtenido de http://www.pmoinformatica.com/2012/11/los-5-valores-de-la-programacion.html

software, I. d. (12 de 04 de 2018). METODOLOGIA XP. Obtenido de http://ingsoftware072301.obolog.es/metodologia-xp-2012877

www.abacovirtual.edu.pe. (12 de 04 de 2018). Obtenido de http://www.abacovirtual.edu.pe/chiclayo/filedocente/140318-CEI2012203MB-000029-19072012-173247.pdf