

Educación Extrema: un paradigma basado en la comunicación docente-alumno

Leo Ordinez

Dpto. de Ing. Elec. y Comp.
Universidad Nacional del Sur - CONICET
Av. Alem 1253, Bahía Blanca, Argentina
lordinez@uns.edu.ar

Manuel Fidel

Dpto. de Cs. e Ing. de la Comp.
Universidad Nacional del Sur
Av. Alem 1253, Bahía Blanca, Argentina
mmf@cs.uns.edu.ar

Resumen

La programación extrema es un método ágil de desarrollo de software que ha tenido una amplia difusión y éxito en aquéllos que la han utilizado. Sus ideas base son diferentes a los métodos tradicionales. Se hace énfasis en aceptar los cambios, en desarrollar de manera evolutiva, en acentuar la comunicación (especialmente entre desarrolladores y clientes), y otras ideas corrientes de la teoría de sistemas. Es notable que un método humanista y sistémico se ofrezca en un campo donde reina el absoluto productivismo. Como se puede comprobar, las ideas introducidas por este método no son exclusivas del campo donde se han originado. Por lo tanto, aquí se propone y explica la utilización de algunos principios y prácticas de este método a la educación, un área que parece necesitar de ellos. Lo expresado se puede continuar ampliando con la elección y adaptación de las partes más útiles del método para la educación.

1. Introducción

El objetivo de este trabajo es presentar un acercamiento entre dos disciplinas aparentemente sin relación: la producción industrial de software y la educación. La primera, marcada históricamente por el afán de lucro y la producción desmedida. La segunda, procurando formar ciudadanos probos que puedan desempeñarse correctamente en los diferentes ámbitos de la sociedad.

En un contexto laboral donde, a excepción del producto final en sí mismo, se involucra la interacción y cooperación entre personas, es necesario y hasta imprescindible tener en cuenta los factores psicológicos, sociológicos y de la vida cotidiana que intervienen en tal entorno. El desarrollo interdisciplinario presupone una mayor amplitud de criterio. Este último implica una apertura mental que golpea fuertemente al método científico clásico, caracterizado por una propuesta básicamente positivista.

El conflicto que provoca el cambio y, en consecuencia, la resistencia siempre hecha a él tiene razones que diferentes escuelas de psicología han explicado desde variados enfoques. Autores como Jean Piaget [18] desde la genética, Sigmund Freud desde el psicoanálisis [11] e incluso B. F. Skinner a partir del conductismo [22], han encarado este interesante tema. Lo cierto es que tal resistencia existe y se manifiesta a menudo.

La sinergia resultante del enfoque interdisciplinario, manifiesta en toda empresa humana, fue el motivo que desencadenó el surgimiento de una variante en la manera de trabajo acostumbrada. En un primer momento, se pensó sólo en un área particular. No es casual que el área particular fuera una ciencia naciente que no ha alcanzado aún una identidad histórica otorgada por el tiempo. Este nuevo enfoque de trabajo sentó sus bases sobre cimientos obviados hasta el momento. El rechazo al cambio fue inminente, se planteaba una reformulación total de lo hecho

hasta el momento en materia de planificación y organización laboral. Cuestiones que surgen del sentido común, del trato que uno pretende para sí por parte de los demás, cuestiones de ética, sentimientos presentes en situaciones de interacción entre personas, aspectos comunes al hecho laboral, diálogos fluidos, principios y valores morales, entre otros, pasaron a conformar la piedra fundacional de un método que al tiempo de su concepción se vio que excedería sus propios límites. Simplemente se invirtió la pirámide convencional, volviendo la vista a la gente que usaría los productos finales, y a aquellos que tienen la misión de construir dichos productos.

El vértigo presente en un mundo competitivo dejó de lado lo fundamental: las personas. También tornando la mirada hacia la materia tangible y al avance tecnológico, y relegando lo básico de la esencia humana. Se pensó en un avance tecnológico sin detenerse a recordar el origen que provocó tal avance. En este punto, es donde el método denominado *eXtreme Programming* (XP) hace aparición como la principal manifestación de los métodos ágiles [2]. Instituye una mirada retrospectiva a la conducta humana y una reformulación de antiguos esquemas; así como un atravesamiento transversal en diversas disciplinas. XP vino a señalar nuevamente el punto de referencia olvidado en el camino del progreso postmoderno. Es un resurgimiento de la frase: *poner la tecnología al servicio del hombre y no al revés*.

Un método con las características de XP puede ser muy útil y fácil de aplicar en cualquier entorno donde participen varias personas. Esta fue la razón fundamental para plantear el acercamiento mencionado en párrafos anteriores. Asimismo, XP ha sido inspiración de una nueva propuesta para la organización industrial por parte de Enrico Zaninotto [24].

A partir de lo expresado, el resto del trabajo se organiza de la siguiente manera: en la sección 2, se presenta el método XP original, junto con algunas ampliaciones necesarias para su posterior aplicación al ámbito educativo; en la sección 3, se abordará la problemática educativa actual y se presentará el enfoque de la Educación Extrema, su ventajas y características; en la sección 4, se plantearán algunos ejemplos sobre la aplicación de la Educación Extrema; y finalmente en la sección 5, se expondrán las conclusiones del trabajo, junto con los próximos pasos a seguir en la línea de investigación del método planteado. El trabajo concluye con la mención de la bibliografía consultada.

2. Orientación Computacional

Los orígenes de XP como método de trabajo apuntaba a las disciplinas relacionadas con la programación de computadoras. En esta sección se hará un repaso y se ampliarán algunos puntos del sistema original, limitado en un principio al entorno de las ciencias de la computación.

2.1. Dificultad de Aceptación y Falsos Conceptos

Se volverá a un tema mencionado en párrafos anteriores por considerarlo fundamental para aclarar dudas que devienen en equívocas primeras impresiones. La resistencia al cambio, presente en las personas, genera dificultades en la aceptación de nuevos conceptos, aún cuando aquellos sean naturales y sencillos. Las bases del método de trabajo expuesto se basan en la observación del trato personal entre diferentes individuos y sobre cuestiones simples del quehacer cotidiano. Pero, es la fuerza que se opone al cambio, al movimiento, quien hace surgir dudas y ataques que dificultan el entendimiento del método y provocan el nacimiento de falsos conceptos.

Uno de las mayores críticas a XP es que no es más que otra versión del método de *prueba y error*. Lo erróneo de esta afirmación se encuentra en pensar a XP como una metodología de desarrollo de software. Cuando, en realidad, es un planteamiento general de una forma de trabajo orientada a lograr satisfacción de los usuarios y los desarrolladores. La generalidad del método conlleva una adaptación a los diversos casos particulares, así como una rápida visión global del proceso de resolución de problemas. No existe ningún inconveniente en acomodar el

método a los requisitos particulares. Más aún, el diseño evolutivo y la adaptación al cambio son premisas fundamentales en XP. Aquí, es precisamente donde se resalta la principal diferencia entre XP y la Ingeniería de Software: mientras el segundo es una metodología de desarrollo, el primero es un enfoque de trabajo general aplicable a diversos campos. Tal es así, que podrían coexistir en un enfoque combinado.

2.2. Ausencia de Modelos Generales

Es común encontrar en la literatura computacional que los diferentes temas planteados tengan como principal sustento teórico el estudio de algún caso particular. Como lo explica Narciso Benbenaste [5], la universalidad de la tecnología computacional permite extremada flexibilidad y versatilidad del sistema debido a su concepción en dos partes antagónicas e interactuantes: el hardware y software. Esta ventaja constructiva que permite el despliegue de numerosas alternativas en la solución de un problema, conlleva una desventaja fundamental: la imposibilidad de construir modelos teóricos universales aplicables de manera práctica a todos los problemas posibles. Existen aproximaciones a dichos modelos, en general, se trata de modelos estrictamente formales, rigurosos y estructurados; inmanejables en problemas de grandes dimensiones. Tampoco se considera, en estos autores, un enfoque sistémico del tema, en cuanto a considerar a todas las partes intervinientes.

La recurrencia a casos especiales en problemas que podrían generalizarse aunque corriendo el riesgo de ser modificados en su concepción, lleva a pensar sobre la utilidad de los modelos generales. Los intentos por englobar diversos problemas y someterlos al control de una metodología que permita resolverlos, suelen terminar en fracasos o en generalizaciones que complican, aún más, la tarea. Aquí se presenta la falencia de los métodos clásicos (y algunos más modernos), de desarrollo de software: intentan construir un modelo para el desarrollo de programas de computación tan general que contempla la mayoría de los problemas comúnmente encontrados y cuya aplicación particular requiere un esfuerzo extra para la construcción de dicho modelo (por ejemplo, UML).

El método general se puede enunciar, pero es necesario un cambio en la óptica de visualización del problema, un reajuste de lo preconcebido en materia de elaboración de software. El enfoque global debe contener la menor cantidad posible de indicaciones específicamente relacionadas con el software. Más aún, la resolución del problema y su codificación son procesos separados y autónomos. Presentan un caos cuidadosamente organizado; en donde, el orden está dado por reglas generales; y la dinámica anárquica del sistema, en pos del objetivo, es quien dictamina el próximo paso a seguir. Se trata de conseguir el fin (un programa que satisface al usuario) a través de los medios que se dispongan. Estos medios, por supuesto, deben estar enmarcados en la legalidad, la honestidad y el respeto por el prójimo.

Por lo tanto, se puede hacer que las bases para la construcción de dicho método se trasladen a otros planos.

2.3. Definición de XP

En esta sección y sus subsecciones, se hará un breve recorrido por el método ágil XP, el cual se compone de cuatro partes construidas en forma creciente sobre las anteriores, siempre evolucionando a partir de conceptos fundamentales, hasta llegar a otros más complejos y elaborados. Asimismo, se intentará ampliar algunas de las partes con el fin de completar el método y poder aplicarlo a otras áreas. Se remite a otros trabajos para ahondar en detalle sobre estos temas [3][4][9][13].



Figura 1: Las cuatro partes que componen XP

2.3.1. Valores

En este tópico y en el siguiente, que tratan de los principios, no se intenta una discusión filosófica sobre la existencia o no de valores y principios morales. Mucho menos, sobre la concepción y origen de los mismos.

A los fines limitados de este trabajo, basta con decir que los valores son cualidades de una realidad ideal presentes en forma diversa en las personas. Los valores se hayan en la vida espiritual y se manifiestan en actos puntuales de la vida cotidiana, sobre todo en las pequeñas encrucijadas.

Los valores propuestos por el método original más algunos agregados se explican a continuación.

Simplicidad No tiene ningún sentido construir complicadas estructuras que dificulten su comprensión y utilización. Es preferible acudir a cosas sencillas y simples que, a primera vista, reflejen lo que se quiere expresar.

Comunicación La mirada se centra en las personas. La característica que diferencia a los humanos de la otras especies es la *palabra*, sobre todo la palabra oral. Es necesario que en un grupo de trabajo haya comunicación para que no se trabaje de más, no se cometan errores y se conserve un ambiente cooperativo.

Retroalimentación Se basa en la generación de preguntas y dudas, cuyas respuestas sirven para adquirir conocimiento del estado del sistema en su conjunto. Involucra también cuestiones no necesariamente técnicas, sino de estados de ánimo y físico de quienes desarrollan o utilizan el sistema.

Coraje En todo emprendimiento, sobre todo en uno novedoso, se necesita tomar decisiones a las cuales, sólo se puede llegar a partir de un acto de valentía. En palabras de Joan Manuel Serrat [21], se trata del heroísmo de *las pequeñas campañas* y de *las pequeñas hazañas*.

Confianza Este método tiene un enfoque hacia el trabajo grupal cooperativo. Por lo tanto, se debe poder confiar en lo que realiza el otro; no sólo en lo que hace cada uno. Si cada miembro del equipo hace su tarea de manera correcta y todos confían en ello, el equipo tendrá resultados exitosos. Esto no excluye la posibilidad de corregir algún error cometido.

Humildad Se refiere a la toma de conciencia de las limitaciones personales y al posible pedido de ayuda ante una dificultad. Es un valor imprescindible al momento de trabajar en grupo. La soberbia envicia al entorno que la padece y determina el fracaso de la empresa.

Perseverancia Ante los escollos y las dificultades, sólo se puede hacer frente con perseverancia y tenacidad. Esto permite continuar en el camino que conduce a la meta, asumiendo con entereza y fuerza las derrotas.

Honestidad Así como los anteriores, este valor es imprescindible en un ambiente cooperativo. Es básico para lograr la colaboración entre personas. La obtención de un buen producto final por medios deshonestos, a la corta o a la larga, se transforma en arma de doble filo y provoca resentimientos y rencores que pueden hacer peligrar el futuro del proyecto.

2.3.2. Principios

Como se explicó anteriormente, las cuatro partes en que se divide XP se basan, a excepción de la primera, en las partes previas. Los principios comienzan a divisar formas de aplicación del método y se vuelven más concretos que los valores. Sin embargo, así como los primeros, se manejan en un plano todavía general, aunque establecen las bases para las actividades y las prácticas.

Los principios, sentados sobre los cimientos de los valores, vienen a dar cuenta del primer paso en la corporización de aquellos. Así como los valores, los principios son formales, en cuanto carecen de contenido concreto; como ejemplifica Omar França-Tarragó en *Introducción a la Ética Profesional* [10] refiriéndose a la frase: *debes evitar dañar al consumidor*. En ella, el mismo autor, aclara que se encuentran presentes dos características comunes en lo que se refiere a principios morales: la asistencia que brindan en cuanto a entendimiento de un proceso, método o sistema: *evitar dañar*; y el apoyo para comenzar la acción a realizar. En este último punto, se ve el paso intermedio entre valores y actividades. Volviendo a la frase expuesta, el contenido explícito se da por: *al consumidor*.

Con todo, es natural que algunos de los principios que se expondrán a continuación sean más aplicables directamente que otros, que se encuentran en un plano de mayor abstracción y dependen de la persona.

A partir de los conceptos previos, se describen brevemente los principios básicos junto con algunos adicionales.

Retroalimentación rápida Se encuentra íntimamente relacionado con la comunicación con el cliente. De nada sirve que dicha comunicación sea esporádica o a intervalos excesivamente espaciados en el tiempo. Lo ideal es que ante el mínimo cambio, realizado por el programador o propuesto por el cliente, el último o el programador, respectivamente, puedan discutir sobre el resultado.

Asumir simplicidad Involucra que el programador resuelva la tarea propuesta lo más sencillamente posible, sin tener en cuenta la previsión sobre cambios futuros. Esto parece contradecir a la expandibilidad del software. Sin embargo, si el diseño y la programación son lo suficientemente sencillos y siguen un estilo de programación clara y basada en estándares, no habrá ningún inconveniente en agrandar el sistema en el futuro. Pero, es en el futuro que se hará la expansión, en consecuencia, por el momento hay que resolver el problema de hoy en forma simple y rápida.

Cambio incremental Los cambios que se deban realizar en el sistema deben ser pequeños, y a medida que se hacen, se deben probar y consultar con el usuario. Este principio, en síntesis, expresa que, en vez de hacer un sólo gran cambio, hay que hacer varios menores. De esta manera, nos acercamos al gran paso de a pequeñas pisadas.

Abrazar el cambio En los métodos de programación *clásicos* no está del todo bien visto que se deban hacer cambios a la mitad del proceso de desarrollo y mucho menos al final. A pesar de esto, la realidad implica un cambio constante y permanente. Todo sistema de

este estilo es evolutivo y dinámico. Por lo tanto, se debe estar preparado física, intelectual y psicológicamente para la aparición de modificaciones que alteren la rutina impuesta. Además, se torna imposible el tener un conocimiento acabado y completo del sistema antes de su construcción. Luego, es de esperar que los pedidos de cambio surjan como consecuencia de la propia dinámica del sistema.

Trabajo de calidad Ésta es una condición *sine qua non*, no se puede comprometer nunca la calidad de un trabajo. Se relaciona fuertemente con tres valores: perseverancia, humildad y honestidad. Si las condiciones del entorno hacen tambalear la entrega de un proyecto a tiempo, es necesario ser honestos y humildes para asumir las responsabilidades y no intentar algún manejo fraudulento de la situación. Asimismo, se debe perseverar aún más para alcanzar la finalización completa: si no se pudo para hoy quizás se pueda para mañana, pero que no sea necesario llegar a pasado mañana. Citando a Pablo Neruda: *Nunca te quejes del ambiente o de los que te rodean, hay quienes en tu mismo ambiente supieron vencer, las circunstancias son buenas o malas según la voluntad o fortaleza de tu corazón.*

Oficio artesanal Mucho se ha discutido sobre la característica artística de la computación. Desde Donald Knuth [15] hasta el presente, han habido confrontaciones para otorgarle el título honorífico de arte al hecho de programar. No es el objetivo de este trabajo emitir una opinión en esa disputa, sino acercarse al acto de programar a un oficio artesanal, donde se requiere de conocimientos básicos, experiencia, constancia y, en particular, mucha inteligencia adaptativa. Es muy sencillo al momento de construir software, caer en tediosos, inentendibles y oscuros lugares de los cuales no se sabe como regresar. Se necesita de habilidad y picardía, a la vez que conocimiento, para resolver problemas de modo elegante y eficiente.

2.3.3. Actividades

En los principios antes enunciados, se comienza a vislumbrar el aspecto dinámico y aplicado del método. En esta sección y en la siguiente tomará forma y sustancia concreta.

Una actividad trae aparejado movimiento y cambio, a la vez que hechos puntuales concretos. Sin embargo, en referencia al último punto, la definición original de XP no las expone como completamente realizables por su carácter aún general. Aquí es donde se manifiesta la *universalidad* del método: quien decida utilizarlo podrá hacerle las adaptaciones que considere pertinentes para su caso particular.

A continuación se detallan las actividades que propone XP.

Escuchar El primer tipo de comunicación humana es la oral. XP toma muy cuenta este aspecto y lo pone en primer lugar en su escala de valores, dejando como secundario la usual comunicación escrita basada en informes y documentos escritos. En general, es más ágil y eficiente el primer tipo de comunicación, ya que involucra una interacción directa entre desarrolladores, o entre desarrolladores y usuarios. Por supuesto que la documentación escrita no está de más, por el contrario, puede ser útil, pero centrarse en que todo esté escrito puede volverse una tarea contraproducente y un desperdicio de tiempo.

Evaluar Este es uno de los puntos más conflictivos de XP. Este se propone diseñar evolutivamente, a partir de lo ya hecho y su correspondiente evaluación. Esto es, generar tests de casos particulares y a partir de ellos comenzar el trabajo de programación. Parece extraño a primera vista. El evaluar antes de programar tiene como principal ventaja que los posibles errores se tendrán en cuenta antes de que aparezcan. En el caso del software, escribir los tests de antemano define los escollos a superar en la siguiente etapa: la programación. Es de destacar que a esta actividad no es fácil acostumbrarse en primera instancia, y requiere

de gran esfuerzo por seguir una tradición radicalmente opuesta, donde los tests se escriben cuando el sistema está completo.

Programar El traspaso de una solución en lenguaje inteligible por personas a uno que lo sea por máquinas, debe seguir conservando el mismo espíritu de comprensión por parte de la gente. Es decir, continuar manteniéndose la comprensibilidad del código a partir del código mismo. Al punto que los comentarios sean innecesarios porque el mismo código habla por sí sólo. En XP, la codificación se basa en la simplicidad.

Diseñar La etapa de diseño, aunque suene contradictorio, está presente durante todo el ciclo de vida del software. Más aún, se plantea que evolucione y crezca por el inherente dinamismo que tienen los sistemas complejos. De este modo, se debe estar preparado para aceptar los cambios y adaptarse a nuevas exigencias que puedan surgir.

Asistir Este es un método de trabajo grupal cooperativo, por lo tanto la asistencia mutua entre los miembros del grupo debe estar presente siempre. Para ello es necesaria una comunicación fluida y un buen ambiente laboral. De aquí, se desprende una de las prácticas de XP que se verán a continuación, la cual expresa que todos pueden conocer el trabajo del resto.

2.3.4. Prácticas

La última de las partes que componen el sistema de trabajo planteado por XP son las prácticas cotidianas que deberían llevarse a cabo. Representan la aplicación de las áreas anteriores. Sin embargo, el conjunto sumado de valores, principios, actividades y prácticas es más que la mera yuxtaposición de todos, genera una nueva dinámica de funcionamiento. El método es el todo, no sus partes.

En este sentido, XP concuerda con la *Teoría General de Sistemas* (ver [6][14]) en aspectos fundamentales de aquella, como son la interacción entre objetos diferentes, los comportamientos emergentes propios de todo sistema (aún más si involucra a personas), la entropía y lo homeostasis. Estas características deben tenerse presente al enfrentar cualquier situación, ya que pueden ser útiles y brindar herramientas que ayuden a enfocar soluciones. Por ejemplo, un conocimiento de la teoría de sistemas proveería el conocimiento para saber que: cuando se enfrenta a un sistema hay partes del mismo que aisladas funcionan de una manera, pero unidas a otras tienen otro comportamiento. O bien, que la información es la única forma de reducir la incertidumbre de un sistema, esto es, la entropía. Para lo cual, es necesario tener un contacto permanente con los usuarios del sistema, a fin de generar una retroalimentación que provoque un decaimiento en esa dirección, esto es, disminuir la entropía.

Así, como ocurrió en los casos anteriores, estas prácticas están sujetas a adaptaciones particulares. Aquí se presentan brevemente.

El juego de planificación Como viene comentándose a través del trabajo, la planificación no juega un papel protagónico en XP. No obstante, no puede dejársela de lado. En este sentido, se debe ser capaz de divisar y bosquejar rápidamente un plan de trabajo de alto nivel para continuar con la siguiente etapa del proyecto.

Pequeñas emisiones El comportamiento aquí se parece al de un periódico de un pueblo, el cual, en vez de publicarse semanalmente con todas las informaciones de la semana, sale diariamente, con menos volumen, pero más actualizado. Estas emisiones favorecen el intercambio con el usuario.

Metáfora En el diálogo entre desarrolladores y usuarios, e incluso entre los mismos desarrolladores, suelen aparecer inconsistencias en el vocabulario y malos entendidos. XP propone que, a partir de dichos diálogos, se genere un *nuevo lenguaje* que concilie a las partes. Se

refiere a construir un puente entre el lenguaje vulgar y el técnico. Por ejemplo, la planificación de procesos que realiza un sistema operativo, es similar a la manera en que se organizan las tareas de limpieza en una casa. O bien, definir un sistema de tiempo real como uno que tiene que *cumplir sus obligaciones en tiempo y forma*.

Diseño simple Sobre este punto se habló en los valores y en los principios. Se sigue manteniendo la misma filosofía: construir pequeñas unidades simples y entendibles.

Evaluar La evaluación es la base del desarrollo en XP. Las pruebas deben construirse primero, a partir del análisis de lo que deberá contemplar cada parte del sistema. Luego, con las pruebas en la mano, se codificará de acuerdo a ellas.

Refactorear Este concepto deriva de la refactorización matemática, en la cual una expresión algebraica se *reacomoda* manteniendo la equivalencia con la original. En tierras informáticas significa mejorar el código ya escrito, pero sin alterar el comportamiento observable del sistema.

Programar en pareja Esta práctica es completamente literal: pretende que dos desarrolladores o un desarrollador y un usuario trabajen juntos en una misma estación de trabajo. Al principio, puede dar la impresión de ser entorpecedor y lento; sin embargo, formando parejas solidarias y que se logren comprender (no necesariamente que piensen siquiera parecido), se lograrán resultados muy interesantes y en poco tiempo. Esto mejora mucho el ambiente laboral. Se puede plantear el caso, en el que las parejas permuten cada cierto tiempo.

Propiedad común Significa que todos los miembros del equipo pueden acceder a ver el trabajo de cualquiera de los integrantes. Incluso, hacer correcciones o sugerir cambios. Por supuesto que, conservando el respeto por la tarea del otro. En esta práctica, asoma un concepto clave en XP: la ausencia de una estructura de liderazgo jerárquica, sobre el que se profundizará más adelante.

Integración continua Las pequeñas emisiones, sobre las que se basa el desarrollo diario de un programa, deben integrarse varias veces al día para evaluar su funcionamiento colectivo e individual.

Semana laboral de 40 horas La sobrecarga de horas en una actividad creativa y que requiere mucha concentración suele ser perjudicial. Más aún, el exceso de trabajo produce resultados mediocres y de baja calidad, trayendo, a largo plazo, graves problemas. Dentro de esta práctica, deben contemplarse ciertos intervalos de descanso periódicos que posibiliten un despejo mental; ya que sin ellos se caería en el mismo terreno que con una carga horaria desmedida.

Cliente *in situ* Sobre este tópico de participación activa del cliente (usuario) del sistema se habló en secciones anteriores. La idea es que el cliente forme parte del grupo de trabajo y pueda realizar los aportes que crea conveniente, porque, en definitiva, es quien va a utilizar el producto final.

Estándares de codificación Se trata de adaptar el trabajo a ciertas estándares internacionales. Esto favorece la compatibilidad del software y la tarea de programación.

2.4. Funcionamiento

Una vez establecidos los conceptos y partes fundamentales del método original, se hará un bosquejo de la interacción de las diferentes premisas planteadas y de su funcionamiento como un todo. Como se resaltó en párrafos anteriores, esta no es la única manera de implementar el

método, quien quiera modificarlo para adaptarlo a sus necesidades particulares lo puede hacer sin inconvenientes (la adaptación a los cambios es primordial en XP).

Los orígenes de XP y sus objetivos a alcanzar lo definen como un método empírico. Esto surge a partir del análisis principalmente de situaciones laborales, donde algunas soluciones presentan un comportamiento acertado en forma teórica, y sin embargo, la práctica muestra lo contrario. De aquí resulta que, los métodos que se reducen a modelos teóricos y formales, se comportan erráticamente. XP promueve solucionar el problema, sin necesidad de encontrar la causa última que lo provocó, ni modificar las partes que funcionan correctamente.

De la definición de XP, se desprende un ciclo de vida para cualquier proyecto, marcado por un fuerte componente de retroalimentación y cambio. En la figura 2 pueden verse la etapas de este ciclo de vida.

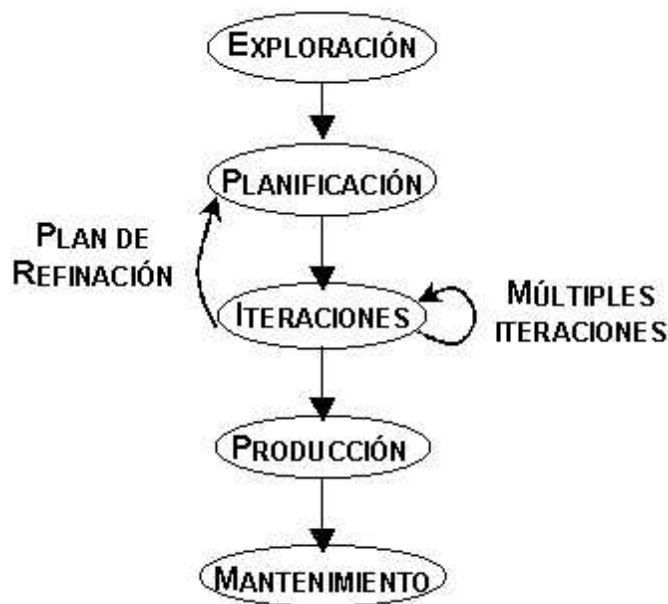


Figura 2: Ciclo de vida de XP

Dentro de cada fase en el desarrollo se caracterizan las siguiente acciones a realizar:

Exploración Comienzo del proyecto, formación del equipo, determinación de los requerimientos generales (de alto nivel) del usuario, construcción de prototipos técnicos.

Planificación Priorización del trabajo y primer plan global.

Iteraciones Evaluación y desarrollo del sistema. Incluye planificación de iteraciones donde ocurren soluciones de necesidades particulares (se baja el nivel a lo específico). Los usuarios finales trabajan aquí en la refinación de cada emisión, asegurando usabilidad.

Producción Despliegue del software al ambiente de producción de los clientes.

Mantenimiento Realización de mejoras y arreglos.

La conformación de un equipo de trabajo está marcada por la confluencia de una gran cantidad de factores que determinan la dinámica de un grupo. En la primer etapa del ciclo de vida se deberán tomar decisiones fundamentales respecto al funcionamiento del grupo. Un grupo de expertos (los mejores), no garantiza resultados exitosos; por esto, generalmente la conformación de un grupo de trabajo está constituido por idóneos en el tema, pero sobre todo, por gente predispuesta a colaborar, y, por supuesto, con conocimientos a la altura del trabajo a realizar.

En las metodologías clásicas se suele categorizar jerárquicamente los roles y establecerlos de modo fijo. XP se deshace de esta estructura fija y confía en la sinergia entre roles dentro del equipo, los cuales no están congelados sino que varían de acuerdo a las circunstancias.

La planificación es una fase corta destacada por marcar el primer acuerdo entre clientes y desarrolladores. Un elemento fundamental que aparece en este punto es la *historia de usuario*. Más adelante, se proseguirá planificando de acuerdo a los segmentos que se decida desarrollar.

El diseño evolutivo que plantea XP permite darle libertad al usuario para definir sus requerimientos en el momento que surjan. Las historias de usuarios vendrían a ser un depósito de requerimientos futuros. Los clientes mantienen un diálogo fluido con el equipo de desarrollo acerca de qué quieren; las historias de usuarios son el punto inicial de esta conversación.

Para concluir lo que se ha expresado, es indispensable detenerse en las iteraciones. Las etapas restantes, o sea *Producción y Mantenimiento*, hablan por sí solas. Sin embargo, el concepto de iteración es novedoso en este ámbito.

Dentro de las *prácticas* vistas en la sección 2.3.4 se destacó el hecho de realizar pequeñas emisiones que puedan ser utilizadas por el cliente. Para lograr ese objetivo, se propone descomponer aún más el problema realizando un refinamiento gradual interno (sin intervención del cliente), hasta alcanzar una de las emisiones mencionadas. En definitiva, si por ejemplo al cliente se le ofrece una nueva emisión cada semana, entonces internamente el equipo de trabajo iterará varias veces desarrollando y corrigiendo errores hasta que esa emisión esté lista dentro del plazo estipulado. Es decir, que las emisiones son entregas del equipo de desarrollo al cliente y las iteraciones son movimientos internos en el proceso de desarrollo de una emisión.

3. Transposición del Método

Como se anticipó en la introducción, el objetivo de este trabajo es ensayar un acercamiento entre un método empírico para desarrollo de software y la actividad educativa. La idea surgió originalmente del análisis del método y de sus enfoques novedosos en cuanto a comunicación entre desarrolladores y clientes. Asimismo, la inserción de ciertos conceptos ya analizados de la Teoría General de Sistemas y su mirada abarcativa y transversal a varias disciplinas, lo volvió aún más atractivo, así como su enfoque humanista y su correcto manejo de grupos. Por otro lado, la elección del área educativa no fue caprichosa. Aunque ya se planteado un acercamiento de estas características (ver [20][23]), sólo se lo ha hecho en referencia a la enseñanza de la programación en el ámbito universitario.

La actualidad argentina, y latinoamericana, está marcada por profundos cambios sociales y desbarajustados enfrentamientos ideológicos, enmarcados en una gran crisis económica que repercute en todas las instituciones. Se está viviendo un momento crítico donde la educación, en general, intenta sostener un modelo insostenible caracterizado por regímenes excesivamente autoritarios, rígidos y dogmáticos. Los regímenes, debido a las crisis sociales, además, tuvieron que mutar a formas institucionalizadas de *contenedores sociales*. Lugares donde se continúa intentando enseñar con un sistema obsoleto, diseñado para una sociedad diferente a la actual; en compensación se brinda apoyo económico, alimenticio, psicológico y espiritual. Este aparente caos educativo requiere de soluciones ágiles que logren alcanzar objetivos necesarios, pero durante el transcurrir de la actividad actual. Y sobre todo, requiere de la comunicación y colaboración de todos los integrantes de la comunidad educativa. Aquí es donde tiene lugar la confluencia del camino de la solución y el del problema.

Como se explicó, cuando se puntualizó sobre el enfoque computacional del método XP (ver sección 2), este no es un método completamente especificado y cerrado, sino que apunta a constituirse como un sistema de trabajo. A partir de esto, al enfocarlo a la educación también debe seguirse la misma senda. La base que se plantea es mantener una comunicación entre el docente y el alumno que permita una fluida retroalimentación. La información proveniente de dicha retroalimentación es sumamente valiosa y útil. Ya que el docente puede determinar de esta manera qué motiva al alumno a realizar una tarea, cuáles temas le parecen más importantes, su

estado de ánimo, su situación grupal, entre otras. A la vez, el alumno lograría mayor participación en su educación y se convertiría en artífice principal de ella y no en un mero receptor de conocimientos. Se generaría un espacio de intercambio y proposición de alternativas para encarar un nuevo tema de estudio, pero con ideas surgidas de los mismos alumnos. Se trata de estimular la creatividad, la elaboración de teorías que, aunque erradas, generan la búsqueda de preguntas y despierten la curiosidad.

Una alternativa con las características mencionadas no es fácil de llevar a cabo, requiere un esfuerzo extra y una visión amplia, pero justificada por los resultados. El momento actual que atraviesa el país está marcado por un fuerte influjo popular y una gran crisis institucional. Se ha perdido la credibilidad en las instituciones primarias de la nación, como son la justicia, la clase dirigente y la escuela no está al margen. Asimismo, la influencia de una protesta popular sin bases sólidas, sin realizar propuestas, contamina la percepción del alumno, el cual demanda su aprobación omitiendo su responsabilidad de aprovechar sus estudios. De lo anterior, se desprende una situación cotidiana de la escuela: se habla de *aprobar*, no de *aprender*. Una de las causas que llevaron a esto es la falta de participación del alumno en la planificación de su educación. En general, no se tiene en cuenta la opinión del alumno en cuanto a la forma en que desea aprender, mucho menos en el contenido que desea aprender.

Por otro lado, la abundante masa de información disponible en la actualidad, pone en tela de juicio la metodología de trabajo tradicional. En ella se incitaba a la recepción pasiva, siendo la escuela el lugar donde se encontraba el conocimiento y los docentes quienes protagonizaban su exposición. El mundo, en estos tiempos, se encuentra inmerso en un huracán informativo, en el cual se mezclan, con escasa posibilidad de distinción, verdades y mentiras, fantasía y realidad, incluso en cuestiones pueriles y de impacto social. La televisión, la radio, los diarios, las revistas y, sobre todo, internet han marcado un punto de quiebre en la demanda de información. La avalancha constante que provocan los medios de comunicación en cuanto a datos, relega a la institución escolar de su lugar de poseedora del conocimiento y presentadora de los saberes. Se quita la sorpresa de la presentación espontánea y se la reemplaza con una segunda visión que carece de atractivo para los alumnos. Estos no son culpables de los bombardeos desmedidos de datos (muchos de los cuales son irrelevantes), a los que se ven sometidos. En este punto, lo peor que se hace es negar la situación.

La búsqueda de conocimiento antes mencionada, se convirtió en una tarea casi trivial, ya que los medios proporcionan tales recursos a discreción. Si a esto se le suma el vértigo característico de estos tiempos, es normal encontrarse con alumnos que no posean interés por las tareas escolares. Se necesita de un método que se pueda adaptar rápidamente a los cambios y provea soluciones en el corto plazo.

Esta propuesta, denominada **Educación Extrema**, tiene como objetivo mostrar un enfoque diferente y ágil para atacar este frente conflictivo que es la educación actual. Se propone un estilo comunicativo, participativo y solidario de enseñanza, haciendo hincapié en la participación del alumno en su educación y en el impulso de una conciencia crítica de la información que se recibe, mediante conocimientos básicos y herramientas de análisis.

3.1. Aplicación Directa

El ciclo de vida de un proyecto de software que se planteó en la figura 2, debe ser adaptado para poder ser aplicado al ámbito educativo. En la figura 3, se muestra la adaptación mencionada. El diagrama expresa los estados generales que se plantean en cuanto a planificación, desarrollo y avance del dictado de una determinada asignatura. A simple vista, se ven tres aspectos determinantes de la Educación Extrema, los cuales son: la comunicación, presente a través de consultas; la evaluación constante y la adaptación al cambio, mostrada por medio del arco que vuelve hacia arriba.



Figura 3: Ciclo de vida escolar

A continuación, se desglosarán cada uno de los componentes del ciclo de vida. En un primer momento, se debe hacer una investigación del estado general del grupo, teniendo en cuenta todos los aspectos que el docente considere relevantes, pero sin dejar de lado los siguientes:

- Situación académica previa. Esto es, los supuestos conocimientos que se debieron haber adquirido en cursos anteriores, de la misma asignatura o afines.
- Metodologías de trabajo usuales que los alumnos conozcan o deseen realizar.
- Características de trabajo grupal, donde se identifique la predisposición a trabajar en conjunto, el tamaño de los grupos, las particularidades del grupo (conveniencia, afinidad, etc.) y la forma de trabajar.
- Características psicológicas del grupo. Identificación de los diferentes componentes del grupo en su totalidad.
- Características socio-económicas, en general.

A partir de esta información, el docente se encuentra en condiciones de realizar una primera aproximación al grupo, a la vez que los alumnos también están realizando dicha aproximación al docente. Como parte de esta primera etapa, denominada **exploración grupal**, puede incluirse, a fin de generar un mayor conocimiento mutuo entre el docente y los alumnos, una presentación por parte del docente de lo que pretende realizar, sus objetivos, conocimientos, formas de trabajo, experiencia previa, entre otros.

Luego de la exploración grupal, la siguiente etapa corre por cuenta únicamente del docente, quien a partir de lo evaluado en la etapa anterior y del plan de estudios curricular, esboza una primera **planificación estratégica**. Aquí, se plantean los objetivos en forma general, se subdivide el trabajo en unidades más pequeñas (de aquí en más, temas), se proponen las primeras metodologías de trabajo para cada tema y se realiza un cronograma tentativo. De lo expresado anteriormente, se puede observar que este es un primer ensayo, que luego se irá adaptando a nuevas exigencias y cambios determinados por la dinámica áulica. En este último punto, es donde hace aparición la retroalimentación propuesta en las bases del método y su correspondiente aspecto comunicacional. Aquí, cobra significado el arco etiquetado como **consulta de refinación**, sobre el cual se especificará más adelante.

A partir de la primer propuesta elaborada en la etapa de planificación estratégica, se realiza un primer intento de acercamiento a un tema particular. Se aplica lo planificado, realizando constantes evaluaciones con el objetivo de determinar el grado de avance en el tema y la aceptación

de la forma de trabajo. La evaluación no tiene porque ser calificativa. Dentro de la etapa **aplicación temática**, existe un punto donde la consulta de refinación resulta fundamental (depende en gran medida de la experiencia del docente), la cual se da cuando se notan situaciones de tensión, ansiedad o malestar general de algunos alumnos. Esta situación puede ser provocada por factores internos al grupo áulico (docente y alumnos) o por factores externos, como puede ser una guerra, por ejemplo.

Tanto si la aplicación temática planteada en un primer instante, se llevó a cabo sin ningún problema o si fueron necesarios reajustes mediante consultas de refinación, una vez alcanzado el objetivo propuesto se regresa a la planificación estratégica para comenzar con un nuevo tema. La llegada al objetivo puede ser parcial con respecto al objetivo original, debido a que una modificación en la planificación estratégica así lo haya dispuesto. Este es el caso en que no se dispone de tiempo, recursos o no se logró encontrar el camino para abordar el tema completo.

4. Ejemplos

En esta sección se brindarán algunos ejemplos representativos de las diversas partes que constituyen el ciclo de vida de la Educación Extrema. En ellos se priorizará el aspecto pragmático, a fin de ilustrar el modo de aplicación del método.

4.1. Matemática (5° EGB)

En primer lugar, en la exploración grupal se realizan las presentaciones personales, de parte del docente y de los alumnos. Se pueden incluir intereses personales (*i.e.* preguntarle a los alumnos qué materia les gusta más). Esto es importante ya que el docente podrá tener idea acerca de la disposición que tendría su grupo a su materia. Esta actividad puede hacerse en forma oral o escrita.

Luego, para conocer los conocimientos acerca de la disciplina (matemática) se realiza una actividad donde se abarquen algunos o todos los temas que están incluidos en el plan curricular. Puede ser un problema abarcativo donde se involucren varios temas diferentes o algún tipo de historia relacionada con la materia.

A partir de ese momento, el docente realizará la planificación, teniendo en cuenta los siguientes aspectos: temas, actividades, recursos didácticos (pizarrón, material concreto, fotocopias, etc.), estrategias didácticas, tiempo, bibliografía. Puede haber otros aspectos que docente tenga en cuenta, pero que no expliciten en la planificación, *i.e.* el diagnóstico inicial del grupo.

En este ejemplo particular el tema a tratar son *fracciones*. Para introducir a los alumnos en el tema, el docente realizará una actividad que consistirá en armar grupos de trabajo, entregarles a cada grupo una hoja de papel y una bolsa con cuatro instrucciones, por ejemplo: *colorear con rojo 4 partes iguales de la hoja*. Las hojas estarán divididas en 8, 12 o 16 partes iguales. Los colores de colores será de cuatro alumnos, habiendo una instrucción para cada uno. Al terminar de realizar la actividad todos los grupos, el docente copiará los resultados de cada grupo conformando una tabla como la del cuadro 1.

Grupo	Parte Azul	Parte Roja	Parte Verde	Parte Amarilla	Suma de las Partes
1	$\frac{5}{16}$	$\frac{6}{16}$	$\frac{2}{16}$	$\frac{3}{16}$	$\frac{16}{16}$
⋮	⋮	⋮	⋮	⋮	⋮

Cuadro 1: Ejercicios de fracciones con colores.

Se copiará en el pizarrón:

En una fracción el denominador indica el número de partes en que se divide la unidad.

El numerador indica el número de partes que se toman de la unidad.

$$\frac{4}{8} \quad \begin{array}{l} \rightarrow \text{Numerador} \\ \rightarrow \text{Denominador} \end{array}$$

Lo dibujamos así...

$$\frac{4}{8} \quad \begin{array}{l} \rightarrow \text{Partes pintadas} \\ \rightarrow \text{Cantidad de partes} \end{array}$$

Luego, se realizarán actividades de fijación y evaluaciones a fin de determinar cómo se incorporó el tema. De esta forma, el docente tendría que volver a lo planificado para reestructurar la planificación estratégica o continuar con el próximo tema.

4.2. Lengua (4° EGB)

La exploración grupal se puede hacer de manera similar al caso anterior, pero puede darse una situación de hostilidad en el grupo, por lo que el docente deberá modificar su reconocimiento del mismo (a través de juegos, encuestas escritas, etc.).

En el caso de esta disciplina, al conocimiento de los saberes puros, se lo puede alcanzar pidiéndoles que produzcan un texto y que trabajen sobre el mismo los conceptos que están establecidos en el curriculum prescripto, es decir, la propuesta oficial que constituye la normativa estructurante de las instituciones educativas, el cual tanto para la institución como para el docente va a ser entendido como un compromiso a cumplir. A partir de esto, se realizará la planificación estratégica (mensual, trimestral o anual) que contendrá los mismos aspectos presentados para el caso de la disciplina matemática (ver 4.1).

La aplicación temática, en este caso, son las *reglas ortográficas*, en especial la *acentuación*, siendo el objetivo fundamental la identificación de palabras agudas, graves y esdrújulas.

El docente, para sondear saberes previos e introducirlos en el tema, repartirá cartones con palabras agudas, graves y esdrújulas (se dividirá a los alumnos en tres grupos cada uno con un tipo de palabra, teniendo cada alumno una palabra). Se les pedirá que separen esas palabras en sílabas, recortando cada una y pegándolas uno a uno en el pizarrón. Luego, le dirá a cada alumno que rodee con una tiza de color la sílaba tónica de su palabra (el concepto de sílaba tónica ya está incorporado). El pizarrón va a estar separado en tres partes y cada grupo (dirigido por la maestra) pegará sus palabras en el color que le corresponda.

Una vez que estén todas las palabras en el pizarrón, el docente pedirá que observen y saquen una conclusión. Aquí es tarea del mismo guiar a los alumnos para que reconozcan el concepto o se aproximen al mismo.

Al tiempo que un alumno reconozca alguna idea, se explicará el tema. Se realizarán actividades de fijación a la vez que se toman evaluaciones no calificativas.

5. Conclusiones y Trabajos Futuros

En el presente trabajo se presentó un paradigma educacional basado en los principios y en el ciclo de vida del método de desarrollo ágil de software denominado *eXtreme Programming* (XP). Con este fin, se realizó una descripción de los principales componentes de XP, junto con algunos secundarios que sirven al propósito de la transposición del método a otra área:

la educación. En el análisis se fue haciendo un diagnóstico (no completo) del estado actual de la educación, de sus problemas y entorno. En base a ese análisis, se ve claramente que el acto educativo es dinámico y cambiante, para lo cual se necesita una forma de trabajo que sea capaz de adaptarse en forma rápida a los cambios. Asimismo, las partes intervinientes en ese acto, docente y alumnos, deben mantener una constante comunicación sobre el progreso del proceso enseñanza-aprendizaje, enfatizando la retroalimentación rápida y también la evaluación permanente y la partición de los temas en secciones manejables. Estas cuestiones se adaptan de modo preciso a las premisas de XP.

De aquí en más, queda un largo camino por recorrer donde este es sólo el primer paso. La educación es un tema vasto, donde todo tipo de aporte para mejorarla es bienvenido. Los próximos pasos a seguir están relacionados con el estudio de casos particulares, donde se aplique el paradigma *Educación Extrema*, a partir de estos estudios se podrá refinar aún más el enfoque propuesto y determinar su grado de aplicación y limitaciones. Además, se ahondará en mayor grado de detalle respecto a cada etapa del ciclo de vida planteado.

Referencias

- [1] I. Aguerrondo, *El planeamiento educativo como instrumento de cambio*, Troquel Educación, Argentina, 1990.
- [2] K. Beck, M. Beedle, A. van. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, *Manifesto for agile software development*, <http://agilemanifesto.org/>.
- [3] Kent Beck, *Una explicación de la programación extrema*, Addison-Wesley, Madrid, España, 2002.
- [4] Kent Beck and Martin Fowler, *Planning extreme programming*, Addison-Wesley.
- [5] Narciso Benbenaste, *Ciencia, tecnología y política*, EUDEBA, 2001.
- [6] L. von Bertalanffy, *Teoría general de los sistemas*, Fondo de la Cultura Económica, México, 1976.
- [7] M. A. Cathalifaud and F. Osorio, *Introducción a los conceptos básicos de la teoría general de sistemas*, Cinta de Moebio (1998), no. 3.
- [8] R. Follari, *Psicoanálisis y sociedad: crítica del dispositivo pedagógico*, no. 16, Lugar, 1997.
- [9] Martin Fowler, *Refactoring: Improving the design of existing code*, Addison-Wesley.
- [10] Omar França-Tarragó, *Introducción a la ética profesional*, Biblioteca Virtual de Etica, Montevideo UCU, 2003.
- [11] Sigmund Freud, *Obras completas*, vol. XXII, ch. Nuevas conferencias sobre introducción al psicoanálisis, Amorrortu, 1932.
- [12] G. Frigerio, M. Poggi, G. Tiramonti, and I. Aguerrondo, *Las instituciones educativas - cara y ceca - elementos para su gestión*, Troquel Educación, Buenos Aires, Argentina, 1992.
- [13] Carlo Ghezzi and Mattia Monga, *Extreme programming: programmazione estrema o revisionismo estremista?*, Mondo Digitale (2002), no. 4.
- [14] Johansen I., *Introducción a la teoría general de sistemas*, Limusa, 1996.
- [15] D.E. Knuth, *The art of computer programming*, vol. I, Addison-Wesley, 1998.
- [16] S. Kovadloff, *Lo posible y lo imposible en educación*, no. 16, Escuela freudiana de Buenos Aires, 1993.
- [17] J. Lacan, *Seminario xi: Los cuatro conceptos fundamentales del psicoanálisis*, Paidós, 1987.
- [18] Jean Piaget, *Seis estudios sobre psicología*, Planeta-Agostini, 1994.

- [19] Alejandro Rozitchner, *No hay conciencia del esfuerzo*, La Nación (2005), no. 48207, 23.
- [20] J. G. Schneider and L. Johnston, *extreme programming at universities - an educational perspective*, Proceedings of the 25th international conference software engineering (Portland, Oregon), 2003, pp. 294–299.
- [21] Joan Manuel Serrat, *Detrás está la gente*, Bienaventurados, Julio 1987.
- [22] B. F. Skinner, *Sobre el conductismo*, Planeta-Agostini, 1994.
- [23] T. H. Wong, *Is extreme programming (xp) a useful tool for computer science education?*, Proceedings of The National Conference on Undergraduate Research (Salt Lake City), University of Utah, March 2003.
- [24] Enrico Zaninotto, *From x programming to the x organization*, 3rd International Conference on Extreme Programming (Alghero, Italia), Mayo 2002.