

البرمجة القصوى أو Extreme Programming

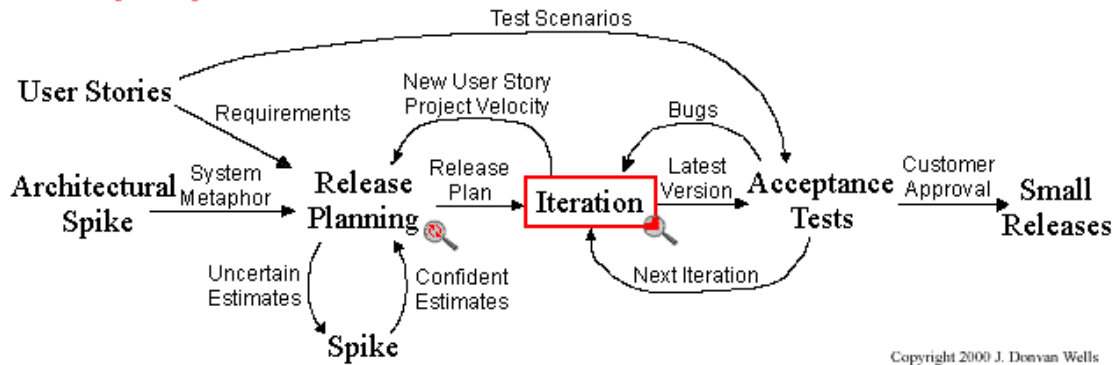
تعتبر البرمجة القصوى أو Extreme Programming أو ما تعرف اختصاراً بـ XP من الطرق الحديثة التي تهتم بتطوير البرمجيات والتي أثبتت جدارتها في العديد من الشركات على اختلاف أحجامها. من أكثر أسباب نجاح البرمجة القصوى XP هي حرصها على تحقيق متطلبات العميل حين الحاجة إليها وهذا بالتالي يعني المرونة في تقبل المتطلبات الخاصة في العميل حتى في مرحلة متأخرة في دورة تطوير المنتج. أيضاً تركز هذه الطريقة على أن يشترك في التطوير جميع الأطراف من فريق العمل والمدراء والعملاء و أيضاً المطورين لأنجاز منتج برمجي ذو جودة عالية .

بأختصار تمثل XP طريقة سهلة و فعالة لتطوير المنتجات البرمجية باستخدام أربع طرق رئيسية هي التواصل، البساطة، التغذية المرجعية، الثبات. يقوم Extreme Programmer بالتواصل مع العملاء من جهة و مع المبرمجين الآخرين من جهة للتأكد من بقاء التصميم سهلاً و نقياً أيضاً يتم إبقاء الباب مفتوحاً للتطوير لاحقاً عن طريق اختبار التصميم و المنتج منذ اليوم الأول و الحرص على تسليمه في أقل وقت ممكن مع القيام بالتعديلات المقترحة و هذا يعطي Extreme Programmer القدرة على الاستجابة للمتطلبات و التقنيات المتغيرة.

إن XP مختلفة عن بقية طرق تطوير المنتجات البرمجية، تشبه في عملها لغز تجميع الصور، حيث أنها تتكون من عدة أجزاء صغيرة ليس لها معنى و لكن عند تجميعها يمكن الحصول على صورة كبيرة يمكن رؤيتها. إنها إنطلاقة كبيرة من طرق تطوير البرمجيات التقليدية إلى طرق حديثة مرنة و قابلة للتطوير.



Extreme Programming Project



Copyright 2000 J. Donovan Wells

يمكن تقسيم مراحل تطوير المنتج باستخدام XP كالتالي :

المرحلة الأولى: التخطيط Planning: وهذه هي النصائح:

- 1- كتابة احتياجات العميل.
- 2- تحديد مراحل لتطوير المنتج.
- 3- تقسيم مراحل التطوير إلى مراحل صغيرة.
- 4- قياس الوقت المتوقع لإنهاء المنتج.
- 5- إعادة التخطيط لكل مرحلة من مراحل التطوير.
- 6- عمل إجتماعات يومية للتخطيط من العملاء و المبرمجين.
- 7- تعديل الخطة حسب المتغيرات الجديدة.

المرحلة الثانية : التصميم Designing :

- 1- التركيز على البساطة.
- 2- استخدام الأصناف Classes والمسؤوليات Responsibilities و التعاون Collaboration بين أعضاء الفريق (الهدف من هذا هو تقسيم العمل بين الفريق كل له مسؤوليات و هذا يبتعد عن طريقة التفكير العادية التي تتولى جميع المهام من قبل شخص واحد فقط).
- 3- انشاء الحلول لتقليل المخاطر (يقصد بهذا أن كل مشكلة تحل لوحدها فقط).
- 4- لا تقم بإضافة الوظائف حالياً ، اهتم الآن بالجدول الزمني و لا تفكر إلا فيما يتوجب عليك إنجازه اليوم.

المرحلة الثالثة : كتابة الكود:

- ١-العميل دائما موجود و هذه هي الميزة في XP حيث أنها لا تجعل العميل متوفرا دائما فقط و لكن تجعله جزءا من الفريق.
- ٢- الكود لا بد أن يوافق المعايير التي تم الإتفاق عليها .(من خلال مرحلة التصميم).
- ٣- كتابة كود إختبار الوحدة البرمجية يساعدك على كتابة الوحدة البرمجية نفسها.
- ٤-كل وحدة برمجية يقوم شخصان فقط بكتابتها حيث يعملان على جهاز كمبيوتر واحد فقط.(السبب أن الشخصين يقومان بزيادة جودة المنتج دون التأثير على زمن الإنتهاء).
- ٥- يقوم شخص واحد فقط بعملية تكامل الوحدات البرمجية (أو فريق بشرط أن يتم التكامل مرة واحدة في الاسبوع لعدم اضاءة الوقت في التكامل).
- ٦- التواصل مع الفريق، حيث يقوم المبرمجون بإتاحة الوحدات البرمجية لبقية الفريق في حال الانتهاء منه و لا يجب ان يبقى أكثر من يوم واحد قبل اتاحه للبقية، الهدف من هذه الخطوة هي التواصل مع اعضاء الفريق لمعرفة الوحدات البرمجية التي يمكن اعادة استخدامها أو التي يمكن مشاركتها.
- ٧- التعاون ، حيث يمكن لأي عضو من الفريق تصميم وحدات برمجية جديدة أو التعديل على الوحدات البرمجية الموجودة لحل مشكلة ما أو إضافة جديد إليها.
- ٨- إترك فكرة تحويله إلى الحل الامثل إلى بعد الانتهاء من كتابة الوحدات البرمجية.
- ٩- لا أوقات إضافية للانتهاء من المنتج البرمجي. يمكنك إعادة تخطيط تحديد مراحل لتطوير المنتج.(انظر المرحلة الاولى النصيحة الثانية).

المرحلة الرابعة : مرحلة الإختبار :

- ١- كل وحدة برمجية لابد أن يتم إختبارها.
- ٢- كل الوحدات البرمجية لابد أن تجتاز الإختبارات قبل أن يتم إعتماها في المنتج.
- ٣- عند إكتشاف ثغرة يتم عمل اختبار لها.
- ٤- عند توصيل المنتج إلى العميل يتم عمل اختبار قبول له. هذا الاختبار يعتمد على متطلبات العميل، يمكن أن يكون للمنتج أكثر من اختبار حسب متطلبات العميل.

خالد بن عبدالله الغنزي