

- بخش اول : انواع متدولوژی های توسعه نرم افزار

متدولوژی های توسعه ی نرم افزار

مقایسه ی متدولوژی های سنگین و چابک

انواع متدولوژی های سبک وزن

- بخش دوم : Extreme Programming (XP)

تاریخچه XP

چرا XP مورد توجه قرار گرفت؟

جایگاه XP در میان دیگر متدولوژی ها

تعریف XP

متدولوژی های توسعه ی نرم افزار به دو دسته تقسیم می شوند :

- متدولوژی های سنگین وزن (Heavy Weight)
- متدولوژی های سبک وزن (Light Weight) که شامل روشهای چابک (Agile) نیز می شوند.

محور اصلی متدولوژی های سنگین وزن مانند RUP

برنامه ریزی جامع و مستند سازی از ابتدا تا انتها و

طراحی کامل و گسترده می باشد.

به عبارتی روشهای

سنگین وزن بصورت پیشگویانه یا Predictive عمل

می کنند یعنی در آغاز همه چیز را پیش بینی

می کنند .

مسلما در ابتدای فازهای تحلیل به دلیل تغییر نیازها

نمی توان همه چیز را پیش بینی کرد بنابراین متدولوژی های سبک وزن بوجود آمدند.

- تمرکز این روش ها بیشتر بر روی سادگی و سرعت است .

این متدولوژی ها را می توان با توجه به معیارهای زیر مقایسه نمود :

✓ روش :

روش های سنگین وزن به صورت پیشگویانه (Predictive) عمل می کنند ولی روش های

چابک سازگار(Adaptive) هستند.

✓ اندازه پروژه :

متدولوژی های سبک وزن بیشتر در پروژه های کوچک استفاده می شوند.

برای پروژه های بزرگ می بایستی از متدولوژی های سنگین وزن استفاده شود.

آمار نشان می دهد که تعداد پروژه های کوچک بسیار بیشتر است!

✓ مدیریت :

در متدولوژی های سنگین وزن مدیریت بصورت مطلق و استبدادی است در حالیکه مدیریت متدولوژی های سبک وزن بصورت غیر متمرکز و آزاد است.

✓ نحوه مستند سازی :

یکی از عیب های متدولوژی های سنگین وزن مستند سازیهای سنگین می باشد که کار بسیار دشوار و زمانبری می باشد و باید بصورت جامع و کامل انجام شود.

✓ در حالیکه در متدولوژی های سبک وزن مستند سازی محدود وبسته به نیاز پروژه انجام می شود.

✓ چرخه های توسعه :

در متدولوژی های سنگین وزن تعداد چرخه ها کم است ولی زمان آنها بسیار زیاد است.

ولی در متدولوژی های سبک وزن چرخه ها بسیار زیاد است و زمان آنها کوتاه.

✓ معیار موفقیت :

در متدولوژی های سنگین وزن معیار موفقیت حرکت در راستای طرح اولیه است.

در متدولوژی های سبک وزن معیار موفقیت بر اساس ارزش کاری (Bussiness Value) مشخص

می شود.

✓ اندازه تیم :

متدولوژی های سنگین وزن احتیاج به یک تیم بزرگ دارند.

در متدولوژی های سبک وزن اندازه تیم کوچک (حداکثر ۳۰ نفر) می باشد.

✓ بازگشت سرمایه : (مهمترین بخش)

در متدولوژی های سبک وزن زودتر به اثر نرم افزار خواهیم رسید که باعث برگشتن سریعتر سرمایه در طول پروژه خواهد شد

✓ در حالیکه در متدولوژی های سنگین وزن باید تا انتهای پروژه صبر کرد !

- این متدولوژی از نوع متدولوژی های چابک (Agile) توسعه ی نرم افزار می باشد.
- اولین بار آقایان Kent Beck, Ward Cunningham و Ron Jefries در شرکت C3 بر روی پروژه ی حقوق و دستمزد در سال ۱۹۹۶ از این متدولوژی بهره گرفتند.
- در سال ۱۹۹۹ آقای Kent Beck کتابی با عنوان Extreme Programming Explained به چاپ رسانید.

■ XP یک متدولوژی Agile برای گروه های کوچک و متوسط است که برای شرایطی که به سرعت در حال تغییر است طراحی شده است.

■ XP متشکل از چهار بخش است : ارزشها, اصول, فعالیت ها و روشها

■ شالوده ی XP ارزش ها هستند که توسط اصول، فعالیت ها و روش ها حمایت می شوند.

*مهمترین چالشی که گروه های نرم افزاری با آن روبرو هستند اطمینان یافتن از رفع نیاز مشتری است زیرا نیازها عموماً غیرواضح و معمولاً متغیرند. پس افزودن قابلیت تغییر و انعطاف پذیری به نرم افزار اهمیت ویژه ای دارد.

*تنها راه تایید اعتبار یک نرم افزار تست کردن , چک کردن با مشتری و اعمال نظرات او در پروژه و تکرار دوباره و دوباره این چرخه است.

چالش های ناشی از گسترش ارتباطات و ورود اینترنت به دنیای کامپیوتر :

- * چگونه سیستم هایی تولید کنیم که بتوانند به سرعت توسعه پیدا کنند؟
- * چگونه در زمان کمتر سیستم تولید کنیم؟
- * چگونه سیستم هایی تولید کنیم که بتوانند با سیستم های دیگر تعامل داشته باشند؟
- * چگونه می توانیم در برابر تغییرات در حال وقوع , واکنش به موقع نشان دهیم؟

* زمانی که نیازهای سیستم به صورت دینامیک تغییر می کند

* زمانی که پروژه دارای ریسک بالاست. برخی ریسک ها شامل:

* تغییر برنامه زمانی

* لغو پروژه

* عدم پذیرش تجاری

* پیچیدگی فنی

* میزان بالای نقص

* کوچک بودن تیم طراحی (کمتر از ۱۰ نفر)

* زمانی که توافق مالی به صورت کامل حاصل نشده باشد.

■ متشکل از چهار بخش است : ارزشها, اصول, فعالیت ها و روشها

■ شالوده ی XP ارزش ها هستند که توسط اصول, فعالیت ها و روش ها حمایت می شوند.

■ XP متشکل از چهار بخش است : ارزشها, اصول, فعالیت ها و روشها

■ شالوده ی XP ارزش ها هستند که توسط اصول, فعالیت ها و روش ها حمایت می شوند.

۱. سادگی (Simplicity) :

یعنی " ساده ترین کاری که ما را به مقصود می رساند انجام

دهیم."

XP به نیاز مشتری در حال حاضر تاکید دارد. در Xp با ساده

فکر کردن, چیزی را که مشتری حقیقتاً نیاز دارد تولید می کنیم, نه آن چیزی را که شاید در آینده به آن نیاز پیدا کند.

در XP هزینه ی تغییرات را ثابت نگه می داریم.

۲. رابطه (Communication) :

XP برای رابطه ی کلامی ارزش بسیاری قائل است.

در این متد بین اعضای همکاری کننده و مشتری رابطه

مستمر وجود دارد.

همچنین از تکنیک هایی چون برنامه نویسی دونفره

(Pair Programming) استفاده می شود.

۳. بازخورد (Feedback) :

محصول با کیفیت حاصل آزمایش و تصحیح خطا از

طریق آن است.

بررسی واکنش مشتری و مصرف کننده درمواجهه با

محصول و اعمال نظرات آن هابسیار حائز اهمیت است.

۴. شهامت (Courage) :

یعنی گفتن واقعیت درباره ی میزان پیشرفت و میزان

درستی تخمین های اولیه, به جای سندسازی برای

پوشاندن شکست ها !

در اینجا جرأتی مورد توجه است که حاصل تفکر عمیق

ودرک درست از مسئله بوده و نتیجه ساده گرایی است

۱. بازخورد سریع (Rapid Feedback) :

بازخورد سریع به معنای ارتباط مستقیم و مکرر با مشتری بعد از انجام هر قسمت، حتی روزانه، و لحاظ کردن بازخورد حاصل در ادامه کار است.

۲. ساده انگاری (Assume Simplicity) :

به این معنی که راه حلی ساده برای هر مسئله پیدا کرده و از بیهوده پیچیده فرض کردن مسائل بپرهیزیم.

۳. تغییر کم کم (Incremental Change) :

در این متد به جای اعمال تغییرات بزرگ و عوض کردن کل پروژه به تبع آنها، با اعمال تغییرات تدریجی و مرتب که در فواصل زمانی کوتاه رخ می دهند به هدف نهایی پروژه نزدیک می شویم.

۴. تغییر سازگار (Embrace Change) :

تغییری است که اعمال می شود و درعین حال که

مشکلی را حل می کند، بقیه ی سیستم را بهم نمی ریزد.

۵. کار با کیفیت (Good Quality) :

روش XP برای بالا بردن کیفیت کاراستفاده از کد، تست شده است.

(۱) شنیدن (Listening) :

XP به جای ارتباط رسمی نوشتاری، صحبت کردن با

مشتری و شنیدن خواست های حقیقی او را مورد توجه

قرار می دهد.

(۲) تست کردن (Testing) :

در XP پیش از کدنویسی، روش مطلوب طراحی شده و

آن را تست می کنند.

(۳) کدنویسی (Coding) :

در این متد کدها به شکلی واضح و قابل فهم وبی نیاز از

وجود توضیح (Comment) نوشته می شوند.

(۴) طراحی کردن (Designing) :

در XP نرم افزار در مرحله ای خاص و جدا از بقیه

مراحل طراحی نمی شود بلکه در طول تولید در حال

طراحی است.

* فعالیت های مربوط به چرخه حیات XP در ۱۲ راه کاربردی توسط XP عرضه می شود

فاز کاوش (Exploration) :

□ کاوش کردن نیازهای کاربر (Exploring User Requirements):

در این مرحله چشم انداز و مأموریت (mission) سیستم شناسایی می شود.

همچنین مشتری ها و تولیدکنندگان با همکاری هم انتخاب های فنی را کاوش کرده , نیازها را تعریف کرده و لیست سناریوهای کاربر را کامل می کنند.

□ نوشتن سناریو های کاربر (Writing User Stories) :

مشتری سناریوهای سطح بالای اولیه را می نویسد و تولیدکنندگان زمان لازم را برای پیاده سازی آن ها برآورد می کنند.

□ برآورد و اکتشاف :

مشتری کشف می کند که چه می خواهد و تولیدکنندگان

روش احتمالی را برآورد می کنند. (تخمین می زنند)

فاز کاوش باعث می شود که تیم مهارت های برآورد

کردنش را بهتر کند.

محصول مرحله کاوش , استعاره ی سیستم ولیست اولیه ای

از سناریوهاست.

□ برآورد و اکتشاف :

مشتری کشف می کند که چه می خواهد و تولیدکنندگان

روش احتمالی را برآورد می کنند.(تخمین می زنند)

فاز کاوش باعث می شود که تیم مهارت های برآورد

کردنش را بهتر کند.

محصول مرحله کاوش ,استعاره ی سیستم ولیست اولیه ای

از سناریوهاست.

برنامه ریزی پروژه (Planning):

در این فاز کوتاه تولیدکنندگان و مشتری روی ویژگی

های (Features) موجود درتحويل اول توافق می کنند.

XPدراین مرحله ازروش بازی برنامه ریزی استفاده میکند.

خروجی این فاز زمان بندی برای یک نشر(Release) است.

چرخه ی تولید (Iteration) :

هر تحويل (Release) به تعدادی چرخه ی تولید تقسیم

می شود که هر چرخه ی تولید حدودا ۱ تا ۴ هفته طول

می کشد.

در این مرحله مشتری سناریوهایی را که می خواهد

انتخاب کرده وتولیدکنندگان ساخت آنها را کامل می کنند.

فرآیند چرخه ی تولید :

❖ تشکیل جلسه برنامه ریزی:

تیم کارراه وظایف (Task) برنامه نویسی تقسیم می کند.

مشتری سناریوهایی را از طرح کلی تحویل ، انتخاب کرده

و اولویت را روی آن هایی که ارزش تجاری بیشتری دارند می گذارد.

فرآیند چرخه ی تولید :

❖ تشکیل جلسه برنامه ریزی:

تیم کارراه وظایف (Task) برنامه نویسی تقسیم می کند.

مشتری سناریوهایی را از طرح کلی تحویل ، انتخاب کرده

و اولویت را روی آن هایی که ارزش تجاری بیشتری دارند می گذارد.

❖ اندازه گیری عملکرد :

XP عملکرد (مقدار کاری که تیم در طول زمان انجام

داده) را اندازه می گیرد.

گروه تولید در قالب تیم های دو نفره روی Task های

مربوط به سناریوها کار می کنند.

هنگامی که زوج های برنامه نویسی وظیفه ای را کامل

کردند روی برد برنامه ریزی علامت می زنند.

❖ تولید و تست کردن :

تولیدکنندگان کار بر روی یک وظیفه را با نوشتن تست برای آن اضافه کردن آن به Test Framework خود آغاز می کنند.

هنگامی که کلاس ها یا کامپوننت ها تمام شدند آنها را به یک ماشین یکپارچه سازی اضافه می کنند.

در انتها مشتری ها تست های قبولی ای که خودشان نوشته اند را بر روی سناریوها اجرا می کنند.

- انواع تست در XP :

■ **Unit Test :**

آزمایشهایی هستند که توسط توسعه دهندگان برای تست کردن Task ها بکار می روند . این آزمایشها پیش از نوشتن هر قسمت کد برای همان قسمت نوشته می شوند.

■ **Acceptance Test :**

این آزمایش برای تست کردن وظیفه مندی کل سیستم توسط مشتریان انجام می شود . این آزمایش معمولاً سرتاسر سیستم یا قسمت بزرگی از آن را در بر می گیرد .

❖ نصب نرم افزار در محیط کاربری :

در تحویل آخر, محصول برای نصب در محیط تولید

مشتری بازبینی و گواهی می شود.

در تئوری، هر سیستمی که با استفاده از XP تولید شده

است را می توان از چرخ ی تولید اول به بعد در محیط

تولید مشتری به کار برد.

❖ نگهداری سیستم بعد از تحویل

بعد از نصب، سیستم وارد وضعیت نگهداری مداوم میشود.

باید استراتژی هایی برای انتقال داده و تعمیر سیستم تعیین

شده باشد.

از پس از اولین تحویل، اعضای گروه باید زمان خود را بین

نگهداری از سیستم به بهره برداری رسیده و تولید بقیه ی

سیستم تقسیم کنند.

XP فعالیت های ذکر شده را در ۱۲ راه کاربردی

عرضه می کند :

(۱) برنامه ریزی (Planning Game) :

یک تعامل محصور (Close Interaction) بین مشتری و

برنامه نویس است. در واقع در این بخش اطلاعات از

مشتری گرفته می شود و برنامه ریزی های لازم انجام

می شود.

(انتشار جزء به جزء و کم کم (Small Releases) :

انتشار دستاوردها ارزش تجاری دارد و چرخه های

XP این امکان را فراهم می کند که در پایان هر مرحله

چیزی برای ارائه داشته باشیم.

(۳) استعاره (Metaphor) :

استفاده از استعاره و تشبیه رابطه ی میان مشتری و

تولیدکننده را تسهیل می کند.

طراحی ساده (Simple Design) :

کد باید ساده ترین راه را برای انجام کار فراهم کند.

راهکارهای XP در این راه :

- اجرای کلیه ی تست ها

- جلوگیری از کدهای تکراری

- تهیه ی مستندات کامل از کلیه ی کدهای نوشته شده

- حداقل کلاس و متد

تست (Testing) :

هسته ی روش XP استفاده از تستهای اتوماتیک است.

- قبل از اینکه برنامه نویس یک خاصیت را اضافه

کند , برای آن یک تست طراحی می کند که بصورت

پیوسته اجرا می گردد .

بازنگری عوامل (Refactoring) :

به معنای بهبود طراحی کدها بدون ایجاد تغییر در رفتار

ظاهری سیستم است.

(V) برنامه نویسی دونفره (Pair Programming) :

به معنای همکاری دوتولیدکننده و تقسیم کردن کارهای

تولید است.

- تولید شامل کدنویسی, فکرکردن, طراحی, تست, گوش

دادن و صحبت کردن است.

مالکیت اشتراکی

در XP اعضا این اجازه را دارند که در هر زمانی

نسبت به تغییر, اصلاح و بهتر کردن کد اقدام کنند.

(۹) یکپارچگی مداوم و پیوسته:

اجزاء سیستم به صورت روزانه ساخته شده و به سیستم

اضافه میشوند.

(هفته ی کاری ۴۰ ساعته(40-hrs Work Week):

XP یک برنامه ی کاری سبک (حداکثر ۸ ساعت کار

در روز) برای اعضای گروه در نظر می گیرد.

(۱۱) حضور مشتری در محل تولید:

نماینده ی مشتری در محل تولید بوده و جزئی از تیم

تولید محسوب می شود بنابراین منافع و خواسته های

مشتری در بالاترین سطح فراهم می شود.