دانشگاه صنعتی شریف

تمرین دوم درس متدولوژی‌های ایجاد نرم‌افزار

استاد:

دکتر رامسین

تهیه کننده:

امید یعقوبی سامانی

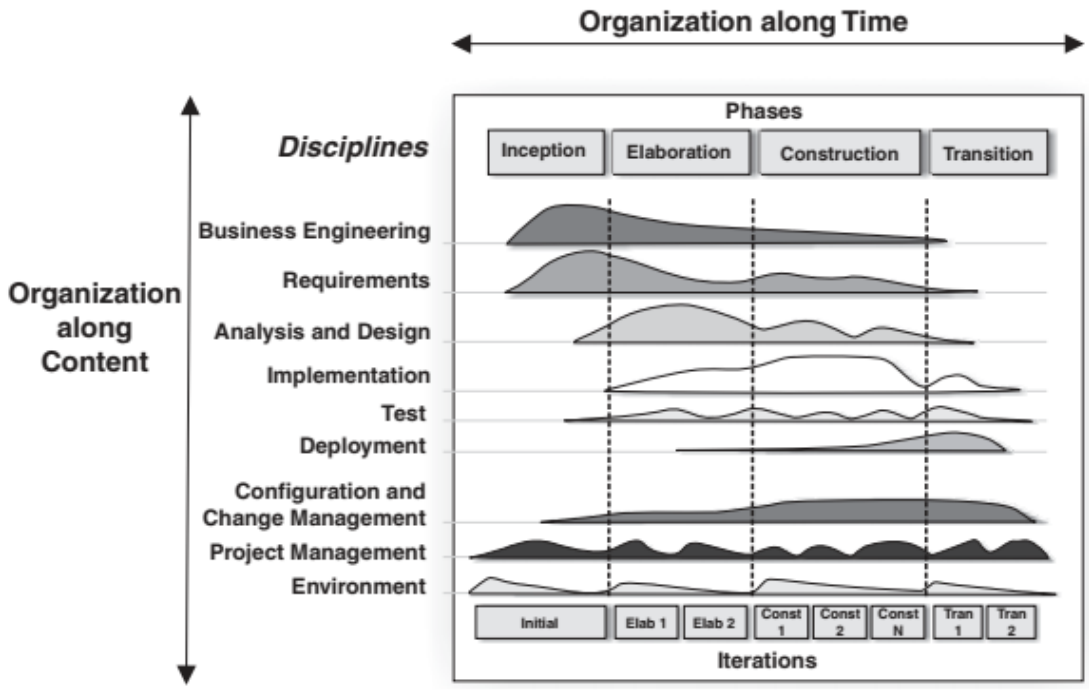
۴۰۱۲۰۵۳۴۸

فهرست مطالب

مقایسه متدولوژی RUP با متدولوژی‌های Fusion، USDP و EUP:

متدولوژی RUP

Rational Unified Process یا RUP یک فرایند مهندسی نرم‌افزار است و ارائه کننده رویکرد منضبط در تخصیص تکالیف و مسئولیت ها در یک سازمان ایجاد است. هدف آن ساخت نرم‌افزار با کیفیت بالایی است که نیازهای کاربران نهایی را در زمان و با بودجه قابل پیش‌بینی تحقق ببخشد. RUP دارای دو بعد است. بعد افقی که شامل فرایندهای چرخه حیات در طول زمان است و بعد عمودی که نظام[۱]های آن هستند و فرایندها را پوشش می دهند. بعد افقی نمایش جنبه های پویای فرایندی که اجرا می شود است که برحسب فاز، تکرار[۲] و موعد[۳] بیان می شود. جنبه عمودی بیان جنبه های ایستای فرایند است که توصیف فرایند برحسب مؤلفه[۴]ها، نظام‌ها، فعالیت[۵] ها، جریان‌های کاری[۶]، مصنوعات[۷] و نقش[۸]ها می باشد.



[1] discipline
[2] iteration
[3] milestone
[4] component
[5] activity
[6] workflow
[7] artifact
[8] role

از لحاظ مدیریتی چرخه حیات در RUP در طول زمان به چهار فاز ترتیبی تقسیم می شود که هر کدام شامل یک موعد بزرگ هستند. در اصل هر فاز یک محدوده زمانی بین دو موعد بزرگ می باشد. در انتهای هر فاز برای تعیین اینکه آیا اهداف فاز محقق شده اند یا نه یک ارزیابی انجام می شود. در صورتی که ارزیابی رضایت بخش باشد پروژه به فاز بعدی منتقل می شود.

۱. *آغاز:* هدف فاز رسیدن به توافق بر سر اهداف چرخه حیات پروژه بین تمام ذینفعان است. این فاز برای یک تلاش جدی ایجاد حائز اهمیت است از این لحاظ که ریسک نیازمندی ها و کسب‌وکار قبل از اینکه پروژه ادامه پیدا کند باید مشخص بشوند. در پروژه هایی که تمرکز روی بهبود سیستم‌های موجود است این فاز مختصر تر است اما باز هم تمرکز روی این است که بررسی شود هم پروژه ارزش اجرا دارد و هم امکان انجام آن هست یا نه.

اهداف اولیه این فاز شامل موارد زیر است:

آ. تعیین قلمرو پروژه و شروط مرزی آن که شامل چشم انداز عملیاتی و معیارهای پذیرش است و می خواهد مشخص کند محصول چه چیزی هست و چه چیزی نیست.

ب. مشخص کردن نیازمندی های بحرانی سیستم که سناریوهای عملیاتی اولیه ای هستند که پیشران trade-off های بزرگ طراحی هستند.

پ. ارائه یا نمایش حداقل یک معماری کاندید برای سناریوهای اولیه

ت. تخمین هزینه و زمان بندی برای کل پروژه و تخمین دقیق برای فاز تفصیل

ث. تخمین ریسک های بالقوه

ج. آماده سازی محیط پشتیبانی برای پروژه

فعالیت های اساسی شامل موارد زیر می باشد:

آ. تدوین قلمرو پروژه

ب. برنامه ریزی و آماده سازی توجیه اقتصادی

پ. تجزیه و تحلیل معماری های کاندید

ت. آماده سازی محیط پروژه

۲. *تفصیل:* هدف این فاز یک معماری تثبیت شده برای سیستم است که بتواند یک پایداری اولیه برای حجم وسیع تلاش های طراحی و پیاده سازی در فاز ساخت فراهم کند. معماری با در نظر گرفتن نیازمندی های مهم یعنی آن نیازمندی هایی که اثر مهمی روی معماری دارند تکامل پیدا می کند و یک ارزیابی برای ریسک است. پایداری معماری از طریق یک یا چند نمونه اولیه ارزیابی می شود.

اهداف اولیه این فاز شامل موارد زیر است:

آ. اطمینان از پایدار بودن معماری، نیازمندی ها و انجام برنامه ریزی به اندازه کافی

ب. نمایش تمام ریسک های معمارانه پروژه

پ. ساخت معماری تثبیت شده

ت. ساخت نمونه تکاملی کامپوننت ها با کیفیت بالا و همچنین یک یا چند نمونه اولیه دور ریختنی برای کاهش ریسک

ث. نمایش پشتیبانی از هزینه و زمان معقول، توسط معماری تثبیت شده

ج. ساخت محیط پشتیبانی

فعالیت های اساسی شامل موارد زیر می باشد:

آ. تعریف، اعتبارسنجی و تثبیت معماری

ب. تدقیق[9] چشم انداز

پ. ساخت و تثبیت برنامه تفصیلی برای تکرارها در فاز ساخت

ت. تدقیق توجیه کننده ایجاد و نصب محیط ایجاد

ث. تدقیق معماری و انتخاب کامپوننت ها

۳. *ساخت:* هدف این فاز شناسایی نیازمندی های باقی مانده و تکمیل ساخت سیستم بر پایه معماری تثبیت شده است. مرحله ساخت به نوعی یک فرایند تولید است که تاکید روی انجام مدیریت منابع و عملیات کنترلی برای بهینه کردن هزینه، زمان بندی و کیفیت دارد. به یک معنا ذهنیت مدیریت از ساخت محصول ذهنی در طی فاز آغاز و تفصیل به ساخت محصول قابل استقرار در فاز ساخت و انتقال تغییر پیدا می کند.

اهداف اولیه این فاز شامل موارد زیر است:

آ. کمینه کردن هزینه ساخت با بهینه کردن منابع و دوری از کارهای غیر ضروری و دوباره کاری

ب. رسیدن به کیفیت کافی

پ. رسیدن به نسخه مفید

ت. کامل کردن تحلیل، طراحی، ساخت و تست همه وظیفه مندی های مورد نیاز

ث. ساخت محصول به روش تکراری افزایشی که آماده انتقال به محیط کاربر باشد

ج. تصمیم اینکه نرم‌افزار، مکان ها و کاربران آمادگی استقرار محصول را دارند یا نه.

[9] refinement

چ. رسیدن به حدی از موازی کاری در کار تیم ایجاد برای افزایش سرعت

فعالیت های اساسی شامل موارد زیر می باشد:

آ. مدیریت منابع، کنترل و بهینه سازی فرایند

ب. تکمیل ساخت کامپوننت ها و تست آن با معیار های ارزیابی تعریف شده

پ. ارزیابی نسخه منتشر شده محصول با معیارهای پذیرش چشم انداز

۴. *انتقال:* تمرکز این فاز روی این است که نرم‌افزار تولید شده در اختیار کاربران نهایی قرار بگیرد و خودش می تواند شامل چندین تکرار باشد. این فاز می تواند شامل تست محصول پیش از انتشار و انجام تغییرات کوچک براساس بازخورد از کاربر باشد. در این نقطه از چرخه عمر بازخورد کاربر روی تنظیم کردن محصول خوب، پیکربندی، نصب و مسائل مربوط به قابلیت استفاده است و مسائل ساختاری خیلی بزرگ باید خیلی زودتر در چرخه حیات حل شده باشند.

در انتهای فاز انتقال هدف چرخه حیات باید محقق شود و پروژه باید در نقطه ای باشد که بتوان آن را بست. در برخی موارد پایان یک چرخه همزمان با آغاز یک چرخه جدید محصول است یا می‌تواند همزمان با تکمیل تحویل مصنوعات به طرف ثالثی باشد که مسئول نگهداری و عملیاتی کردن سیستم است. این فاز بسته به نوع پروژه می تواند خیلی سر راست یا خیلی پیچیده باشد و فعالیت های انجام شده در آن بسته به هدف آن دارد.

اهداف اولیه این فاز شامل موارد زیر است.

آ. تست بتا برای تایید سیستم جدید با انتظارات کاربر

ب. تست بتا و عملیاتی شدن موازی با سیستم موروثی که قرار است با آن جایگزین شود

پ. تبدیل پایگاه داده عملیاتی

ت. آموزش کاربران و نگهداری کنندگان سیستم

ث. ارائه به نیروهای فروش، توزیع و بازاریابی

ج. مهندسی خاص استقرار مانند بسته بندی تجاری، تولید، فروش و آموزش کارمندان میدانی

چ. فعالیت های تنظیمی مثل رفع باگ، بهبود کارایی و قابلیت استفاده

ح. ارزیابی محصول استقراریافته با چشم انداز و معیارهای ارزیابی پروژه

خ. رسیدن به خود پشتیبان بودن کاربر

د. رسیدن توافق ذینفعان روی اینکه محصول استقرار یافته کامل است

ذ. رسیدن توافق ذینفعان روی اینکه محصول استقرار یافته سازگار است با کمک ارزیابی معیارهای چشم انداز

فعالیت های اصلی شامل موارد زیر است:

آ. اجرای فاز استقرار

ب. نهایی کردن موارد پشتیبانی کاربر

پ. تست محصول در محیط ایجاد

ت. ساخت نسخه انتشار محصول

ث. گرفتن بازخورد کاربران

ج. تنظیم محصول براساس بازخوردها

چ. قابل دسترسی کردن محصول برای کاربران نهایی

هر تکرار در فرایند شامل ۹ واحد کاری است که به آنها نظام می گویند. این نظامات شامل موارد زیر است:

۱. *مدل سازی کسب‌وکار:* توصیف فرایندهای کسب‌وکار و ساختار داخلی کسب‌وکار برای فهم آن و تعیین نیازمندی های سیستم نرم‌افزاری که باید ساخته شود هدف این نظام است که در نتیجه این فعالیت ها Business Usecase Model و Business Object Model ساخته می شود.

۲. *مدیریت نیازمندی ها:* دغدغه این نظام استخراج، سازماندهی و مستندسازی نیازمندی ها است و در نتیجه آن Usecase Model ساخته می شود.

۳. *تحلیل و طراحی:* دغدغه، ساخت معماری و طراحی سیستم نرم‌افزاری است. نتیجه این نظام مدل طراحی است. مدل طراحی تولیدش اختیاری است. مدل طراحی شامل کلاس های طراحی است که با پکیج های طراحی ساخت یافته شده اند و زیر سیستم ها و اینترفیس هایی هستند که به خوبی تعریف شده اند تا نمایش دهنده چیزی باشند که در زمان پیاده سازی به کامپوننت تبدیل می شود. همچنین شامل توصیفات چگونگی همکاری اشیا این کلاس ها برای تحقق Usecase ها است.

۴. *پیاده سازی:* هدف نوشتن و دیباگ کد منبع است. همچنین انجام unit test و مدیریت build محصول. فایل های کد منبع، فایل های قابل اجرا و فایل پشتیبان در نتیجه این نظام تولید می شوند.

۵. *تست:* دغدغه تست های یکپارچه سازی، سیستمی و پذیرش است.

۶. *استقرار:* هدف بسته بندی نرم‌افزار، ساخت اسکریپت نصب، نگارش مستندات کاربر نهایی و تکالیف مورد نیاز برای اینکه نرم‌افزار در اختیار کاربر نهایی قرار بگیرد است.

۷. *مدیریت پروژه:* هدف برنامه ریزی، زمان بندی و کنترل است.

۸. *مدیریت تغییرات و پیکربندی:* دغدغه مدیریت نسخه ها، انتشارها و مدیریت درخواست تغییرات است.

۹. *محیط:* دغدغه و انطباق فرایند با نیازهای پروژه یا سازمان و معرفی و پشتیبانی از ابزارهای ساخت.

متدولوژی USDP

Unified Software Development Process یا USDP یک فرایند مهندسی نرم‌افزار است که معمولا با نام Unified Process یا UP شناخته می شود.UP بر پایه فرایندهای کاری در شرکت های Ericson، Rational و منابع دیگر رویه های عملی می باشد.

RUP نسخه ای تجاری از UP می باشد که در سال ۲۰۰۳ در شرکت Rational ساخته شد و شامل ابزارها، استانداردها و ملزومات دیگر است که در UP وجود ندارد و برای استفاده از UP باید آنها را فراهم کرد. UP یک فرایند عمومی ایجاد نرم‌افزار است که باید برای هر سازمان و هر پروژه خاص آن نمونه سازی و سفارشی شود. UP این را که همه پروژه های نرم‌افزاری مقاصد مختلف دارند و اینکه رویکرد تک سایز برای همه کار نمی کند را به رسمیت شناخته است. فرایند نمونه سازی شامل تعریف و گنجاندن موارد زیر است:

۱. استانداردهای داخلی
۲. قالب مستندات
۳. ابزارها، کامپایلرها و ابزارهای مدیریت پیکربندی
۴. پایگاه داده، رهگیری باگها و رهگیری پروژه
۵. تغییرات چرخه عمر (مثلا برای پروژه های با بحرانیت بالاتر معیار های کنترل کیفیت باید پیچیده تر باشند)

با اینکه RUP کاملتر از UP است حتی آن هم با همین روش برای پروژه های مختلف سفارشی سازی و نمونه سازی می کند اگرچه کار مورد نیاز برای انجام آن خیلی کمتر از شروع با UP خام است. با هر روش مهندسی نرم‌افزاری باید مقدار مشخصی هزینه و زمان صرف نمونه سازی و سفارشی سازی فرایند برای پروژه بشود.

UP دارای سه اصل اساسی است:

۱. مبتنی بودن بر نیازمندی ها و ریسک
۲. معماری محور

۳. تکراری و افزایشی

چون روشی برای جمع آوری نیازمندی ها دارد می توان به طور دقیق گفت مبتنی بر نیازمندی ها است. ریسک پیشران دیگر UP است که اگر به طور فعال به ریسک ها حمله نشود آنها حمله می کنند و UP این را با پیش بینی ساخت نرم‌افزار با تحلیل ریسک ها نشان می دهد.

رویکرد UP برای ساخت سیستم نرم‌افزاری، ساخت و تکمیل یک معماری سیستم مستحکم است. معماری توصیف جنبه استراتژیک چگونگی شکستن سیستم به کامپوننت ها، ارتباط و تعامل کامپونت‌ها و نحوه استقرار آنها روی سخت افزار است. به طور مشخص یک معماری سیستم با کیفیت، به سیستم با کیفیت ختم می شود.

UP تکراری و افزایشی است. جنبه تکراری بودن به این معنی است که پروژه به زیر پروژه های کوچکتر (تکرار) شکسته می شود که وظیفه مندی های سیستم را به صورت تکه ای یا افزایشی تحویل می دهند که در نهایت منجر به وظیفه مندی کامل سیستم خواهد شد. به عبارت دیگر نرم‌افزار با یک فرایند قدم به قدم تدقیق، به هدف نهایی می رسد. نکته کلیدی این است که هر تکرار شامل همه عناصر عادی ایجاد پروژه نرم‌افزاری است که شامل برنامه ریزی، تحلیل و طراحی، ساخت، یکپارچه سازی و تست و انتشار داخلی یا خارجی می باشد.

هر تکرار خط مبنایی[10] تولید می کند که شامل بخشی از سیستم نهایی و تمام مستندات مرتبط با پروژه است. خط مبنا انتشار داخلی یا خارجی مجموعه ای از مصنوعات بازبینی شده و تایید شده بوسیله آن تکرار است. هر خط مبنا یک پایه توافق شده برای بررسی و ایجاد بیشتر فراهم می کند و نقطه تایید شده و ثابت زمان بندی، بودجه و محدوده پروژه است و تنها می تواند از طریق فرایند رسمی مدیریت تغییر و پیکربندی، تغییر کند. این خطوط مبنا روی هم و در تکرارهای متوالی ساخته می شوند تا سیستم نهایی تولید شود. تفاوت بین دو خط مبنای متوالی به عنوان افزایش شناخته می شود برای همین است که چرخه حیات UP تکراری و افزایشی شناخته می شود.

هر تکرار شامل ۵ جریان کاری می باشد:

۱. نیازمندی ها: جمع آوری چیزی که سیستم قرار است انجام بدهد.
۲. تحلیل: تدقیق و ساختارمند کردن نیازمندی ها

[10] baseline

۳. طراحی: تحقق نیازمندی ها در معماری سیستم
۴. پیاده سازی: ساخت نرم‌افزار
۵. تست: تایید کارهای پیاده سازی به طور خواسته شده

اگرچه هر تکرار ممکن است شامل هر ۵ جریان کاری اصلی باشد اما تاکید روی یک جریان کاری خاص بستگی به جایی که تکرار در چرخه حیات رخ می دهد دارد.

شکستن پروژه به توالی تکرارها اجازه یک رویکرد انعطاف پذیر برای برنامه ریزی پروژه را می دهد. ساده ترین رویکرد ترتیب زمانی توالی تکرار ها است که به ترتیب اجرا می شوند. اگرچه گاهی تکرارها می توانند موازی اجرا شوند که بستگی به وابستگی بین مصنوعات هر تکرار و نیازمند رویکرد خاص در معماری و مدل سازی است. مزیت تکرارهای موازی در کاهش زمان ورود به بازار و استفاده بهتر از تیم ها است که البته باید با دقت برنامه ریزی شود.

چرخه حیات پروژه دارای چهار فاز است:

۱. *آغاز:* در این فاز امکان سنجی انجام می شود که می تواند شامل نمونه سازی فنی برای تایید تصمیمات تکنولوژیکی یا نمونه سازی proof of concept برای اعتبار سنجی نیازمندی های تجاری باشد. همچنین توجیه اقتصادی برای نمایش اینکه پروژه دارای منفعت تجاری قابل اندازه گیری است ساخته می شود و نیازمندی های اساسی برای تعیین قلمرو سیستم جمع آوری می شوند. ریسک های اساسی هم باید اینجا شناسایی بشوند.
۲. *تفصیل:* در این جا یک معماری تثبیت شده قابل اجرا ساخته می شود. ارزیابی ریسک ها دقیق شده و به آنها جزئیات کافی اضافه می شود. usecase ها تا ۸۰ درصد جمع آوری می شوند. برنامه تفصیلی برای فاز ساخت ایجاد می شود و یک پیشنهاد شامل منابع، زمان بندی، تجهیزات، افراد و هزینه ها تدوین می شود. هدف اصلی این فاز ساخت معماری تثبیت شده قابل اجرا است که طبق مشخصات معماری ساخته شده و دور ریختنی نیست و این معماری در ادامه فرایند تکمیل و به سیستم نهایی تبدیل خواهد شد.
۳. *ساخت:* در این فاز همه نیازمندی ها، تحلیل ها و طراحی ها تکمیل می شوند. معماری تثبیت شده فاز تفصیل اینجا تکمیل و سیستم نهایی ساخته می شود. مسئله اصلی در اینجا یکپارچه نگهداشتن معماری سیستم است و اینکه فشار روی تیم ها باعث کاهش کیفیت سیستم نهایی و در نتیجه افزایش هزینه نگهداری شود رایج است.

۴. *انتقال:* در این فاز تست بتای سیستم کامل و سیستم نهایی استقرار پیدا می کند. خطاها و ایرادات رفع می شوند. محیط کاربر برای نرم‌افزار جدید آماده می شود. نرم‌افزار برای عملیاتی شدن در محیط کاربر تنظیم می شود. در صورت بروز مشکل در سیستم تغییرات لازم انجام می شوند. راهنمای کاربری و دیگر مستندات مورد نیاز ایجاد یا تکمیل می شوند. به کاربران مشاوره داده می شود و بررسی های پسا پروژه ای انجام می پذیرد.

هر کدام از فازها با یک موعد بزرگ پایان می پذیرد. در داخل هر فاز می توان یک یا چند تکرار داشت و در هر تکرار ۵ جریان کاری اصلی و جریان های کاری اضافی اجرا می شوند. تعداد دقیق تعداد تکرار در هر فاز بستگی به سایز پروژه دارد. اما هر تکرار نباید بیش از ۲ تا ۳ ماه طول بکشد.

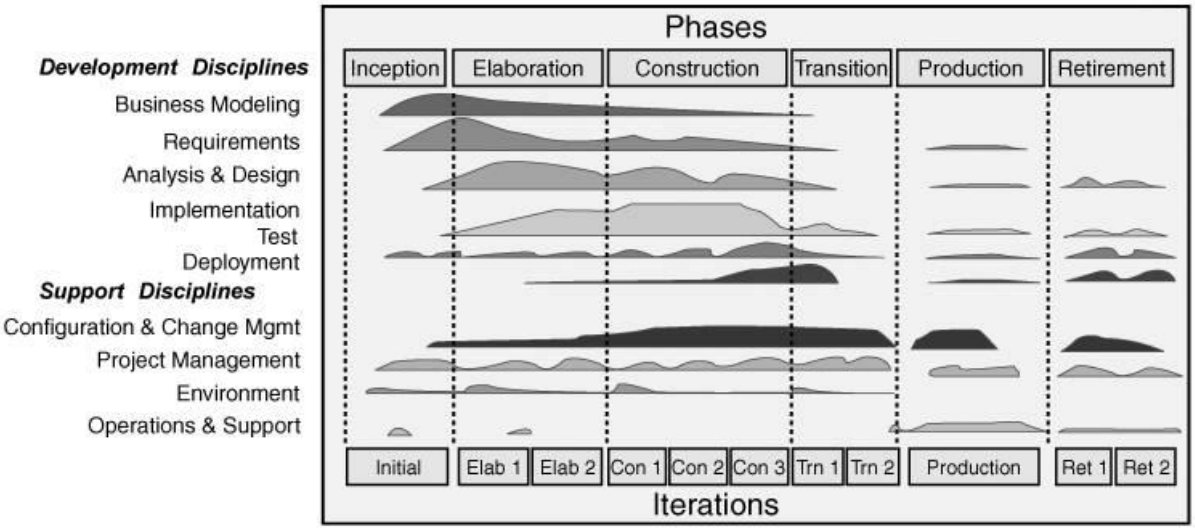
یکی از ویژگی های بزرگ UP این است که به جای اینکه فرایند مبتنی بر تحویل دادنی باشد مبتنی بر هدف است و هر فاز دارای یک موعد شامل مجموعه شروط رضایت است و اینکه شروط شامل ساخت تحویل دادنی های خاص باشند بستگی به هدف مورد نیاز پروژه دارد.

متدولوژی EUP

Enterprise Unified Process یا EUP در سال ۱۹۹۹ معرفی شد و بعدها بهبود پیدا کرد. EUP گسترش یافته RUP می باشد و هر دو RUP و EUP نمونه سازی شده از UP می باشند. اگرچه RUP چرخه ایجاد نرم‌افزار را تعریف کرده است EUP آن را گسترش داده تا بتواند چرخه کامل فناوری اطلاعات را پوشش بدهد. چرخه کامل سیستم از نقطه ای است که شخصی ایده ای را مطرح می کند تا زمان بازنشسته شدن آن ادامه دارد. بخش مهم چرخه، ایجاد اولیه و انتشار در محیط کاربر همچنین تکامل مداوم آن و نگهداری در طول زمان است.

EUP با گسترش RUP به آن عملیات و پشتیبانی سیستم بعد از ورود به محیط کاربر و بازنشستگی سیستم را اضافه نموده است. به علاوه چون اکثر سازمان ها غیر از سازمان های کوچک بیش از یک سیستم دارند EUP می تواند مسائل بین سیستمی مثل مدیریت پورتفو، معماری سازمانی و استفاده مجدد استراتژیک را هم مدیریت نماید. در حالیکه RUP یک چرخه خیلی خوب برای ایجاد نرم‌افزار تعریف می کند EUP نمایش دهنده کل چرخه فناوری اطلاعات است.

EUP چرخه حیات RUP را با اضافه کردن نظام عملیات و نگهداری همراه با دو فاز جدید تولید[11] و بازنشستگی[12] گسترش داده است. با انجام این کار، اینها روی چرخه حیات منعکس شده و از ساخت نسخه اولیه تا پشتیبانی و بازنشستگی نهایی را پوشش خواهد داد.



تولید:

به عنوان پنجمین فاز اضافه شده و تمرکز این فاز روی نگهداری نرم‌افزار در حالت عملیاتی تا زمانی که با نسخه جدید جایگزین شود (با اجرای دوباره چرخه از ابتدا)، یا بازنشسته شود یا حذف شود است. در طی این فاز تکرار نداریم. این فاز چیزی شبیه فاز نگهداری در چرخه عمومی ایجاد نرم‌افزار است و دغدغه اصلی آن عملیات و پشتیبانی سیستم است اما بر خلاف نگهداری کلاسیک هر نیازی به تغییر سیستم حتی رفع باگ نیازمند اجرای دوباره چرخه ایجاد است.

بازنشستگی:

در این فاز که در سال ۲۰۰۲ به عنوان ششمین فاز اضافه شده تمرکز روی حذف سیستم از محیط عملیاتی است چون یا دیگر به سیستم نیازی نیست یا باید جایگزین شود و معمولا شامل موارد زیر است:

آ. شناسایی ارتباط سیستم با سیستم های دیگر

[11] production
[12] retirement

ب. بازطراحی و کار دوباره روی سیستم های دیگر به طوریکه که دیگر وابسته به سیستمی که قرار است بازنشسته شود نباشند.

پ. تبدیل اطلاعات موروثی جاری

ت. بایگانی کردن اطلاعاتی که توسط سیستم نگهداری می شود و در سیستم جاری دیگر نیازی به آنها نیست.

ث. مدیریت پیکربندی سیستم حذف شده به طوریکه که بتواند اگر درآینده لازم بود دوباره نصب شود

ج. تست system integration سیستم های باقیمانده برای اطمینان از اینکه با حذف این سیستم آنها دچار نقص نشده باشند

به EUP دو نظام جدید نسبت به RUP اضافه شده است:

عملیات و پشتیبانی:

هدف مسائل مربوط به عملیاتی نگه داشتن و پشتیبانی سیستم است و نوعا مرتبط با فاز نگهداری در چرخه ایجاد نرم‌افزار می باشد. این نظام در چندین فاز دیگر غیر از تولید گسترش یافته است. در طی فاز ساخت و احتمالا ابتدای تفصیل ساخت برنامه پشتیبانی و عملیاتی و مستندات و راهنماها آغاز می شود. این مصنوعات بهبود یافته و در فاز انتقال تکمیل می شوند که اینجا شامل آموزش افراد برای عملیاتی نگه داشتن و پشتیبانی سیستم است. در فاز تولید و بازنشستگی این نظام پوشش دهنده فعالیت های کلاسیک نگهداری است.

مدیریت سازمانی:

هدف فعالیت های مورد نیاز ساخت، تکمیل و نگهداری محصولات سیستمی سازمان مثل مدل های در سطح سازمان(نیازمندی ها و معماری)، فرایند نرم‌افزار، استانداردها، راهنماها و مصنوعات قابل استفاده مجدد است. این نظام خودش شامل ۷ نظام است:

۱. مدل سازی کسب‌وکار سازمانی
۲. مدیریت پورتفو
۳. معماری سازمانی
۴. استفاده مجدد استراتژیک
۵. مدیریت منابع انسانی
۶. مدیریت سازمانی

۷. بهبود فرایند نرم‌افزار

متدولوژی Fusion

متدولوژی Fusion اولین بار در سال ۱۹۹۲ توسط تیمی در آزمایشگاه HP معرفی شد. نسخه بازبینی شده و تفصیلی آن در سال ۱۹۹٤ منتشر شد. متدولوژی نتیجه ادغام، یکپارچه سازی و گسترش تعدادی متدولوژی قدیمی مثل OMT، Booch، Objectory و RDD بود برای همین هم Fusion نام گرفت. طراح متدولوژی گفته است که چرخه متدولوژی را از نیازمندی ها تا پیاده سازی پوشش می دهد گرچه فاز تحلیل زمانی آغاز می شود که نیازمندی های اولیه آماده باشند و این نیازمندی ها در واقع ورودی اصلی کل فرایند می باشند. Fusion بررسی سازگاری و کامل بودن بین فازها را برای امکان پذیر کردن پیشرفت منظم و قابل اعتماد در مراحل ایجاد نرم‌افزار فراهم می کند. همچنین معیارهایی را برای تعیین زمانی که باید از یک فاز به فاز بعدی چرخه متدولوژی رفت پیشنهاد می دهد.

فرایند Fusion دارای سه فاز است:

۱. *تحلیل:* تمرکز روی بررسی چیستی سیستم است. سیستم از دیدگاه کاربر توصیف می شود. نیازمندی های سیستم به مشخصات سیستم نگاشت می شوند که از طریق مجموعه ای از مدل ها بیان می‌شوند. مدل هایی که در این فاز ساخته می شوند توصیف کننده موارد زیر هستند:
   آ. اشیا و کلاس های قلمرو کاربرد و رابطه بین کلاس ها و اشیا
   ب. عملیاتی که توسط سیستم اجرا می شود.
   پ. ترتیب مناسب این عملیات
   در این فاز Transaction Scenario، System Interface Diagram، System Object Diagram، Life Cycle Model و Operation Schema تولید می شود.
۲. *طراحی:* در این فاز تمرکز روی چگونگی کار سیستم تعریف شده در فاز تحلیل است. مشخصات سیستم به عنوان خروجی فاز قبل به طراحی برای پیاده سازی سیستم نگاشت می شود. مدل های تولید شده در این فاز موارد زیر را توصیف می کنند:
   آ. تحقق عملیات سیستم از لحاظ همکاری اشیا
   ب. چگونگی ارتباط اشیا با هم
   پ. چگونه کلاس ها که اشیا به آنها تعلق دارند از طریق ساختار توارث کلاس ها اختصاصی و تدقیق می شوند.

ت. جزئیات دقیق attribute ها و method های کلاس ها

در این فاز Object Iteration Graph، Visibility Graph، Inheritance Graph و Class Description تولید می شود.

۳. *پیاده سازی:* تمرکز روی کد زنی سیستم است. طراحی سیستم به محیط خاص برنامه نویسی نگاشت می شود. کلاس های طراحی به کلاس های خاص زبان و ارتباط اشیا به method ها پیاده سازی می شوند.

در سال ۱۹۹۷ متدولوژی با یک بروزرسانی فعالیت هایش را بروز کرد و فازهای نیازمندی ها و معماری را به سه فاز دیگر هم اضافه نمود. همچنین فعالیت های مدیریت فرایند هم برای آن تعریف شده است.

در ادامه این متدولوژی ها بررسی و مقایسه بین آنها صورت می گیرد.

معیارهای فرایند:

معیار تعریف:

متدولوژی باید به خوبی تعریف و مستند شده باشد و مستند آن باید توصیفی جامع، شفاف، منطقی، دقیق، تفصیلی و سازگار ارائه کند.

چرخه عمر و واحدهای کاری:

متدولوژی ***RUP***: چرخه عمر متدولوژی شامل چهار فاز آغاز، تفصیل، ساخت و انتقال است که صورت ترتیبی اجرا می شوند و درون هر فاز به صورت تکراری انجام می شود. دارای ۹ نظام است که کل چرخه را پوشش می دهند اما تاکید روی هر کدام بستگی به هدف فاز دارد.

متدولوژی ***Fusion***: چرخه عمر متدولوژی شامل سه فاز تحلیل، طراحی و پیاده سازی می باشد که به صورت ترتیبی اجرا می شوند.

متدولوژی ***USDP***: چرخه عمر متدولوژی شامل چهار فاز آغاز، تفصیل، ساخت و انتقال است که صورت ترتیبی اجرا می شوند و درون هر فاز به صورت تکراری انجام می شود. دارای ۵ نظام است که کل چرخه را پوشش می دهند اما تاکید روی هر کدام بستگی به هدف فاز دارد.

متدولوژی EUP: چرخه عمر متدولوژی شامل شش فاز آغاز، تفصیل، ساخت، انتقال، تولید و بازنشستگی است که صورت ترتیبی اجرا می شوند و درون هر فاز به صورت تکراری انجام می شود. دارای ۱۷ نظام است که کل چرخه را پوشش می دهند اما تاکید روی هر کدام بستگی به هدف فاز دارد.

مقایسه RUP با Fusion: تعریف فازها و واحد های کاری در RUP بسیار دقیق تر از Fusion است.

مقایسه RUP با USDP: چون RUP گسترش یافته UP است و UP سطح بالاتر است تعریف فازها و تکرارها در RUP دقیق تر و بهتر از UP می باشد.

مقایسه RUP با EUP: تعریف فازها و تکراری این دو تا متدولوژی از نظر دقت جزئیات مشابه هم هستند چون EUP گسترش یافته RUP می باشد و تفاوت آنها در فازهای اضافه شده به EUP است.

نقش ها و افراد:

متدولوژی RUP: در این متدولوژی نقش ها به طور کامل و جامع همراه با شرح وظایف و مصنوعاتی که در ساخت آن دخیل هستند مشخص شده است. در این متدولوژی ۳۰ نقش وجود دارد که در ۵ دسته اصلی تحلیلگر، توسعه دهنده، مدیر، تست کننده و تولید و پشتیبانی دسته بندی شده اند.

متدولوژی Fusion: در این متدولوژی اشاره صریحی به نقش ها و افراد دخیل در آن نشده است.

متدولوژی USDP: در این متدولوژی نقش ها در حد خوبی تعریف شده اند و مصنوعاتی که در ساخت آن دخیل هستند هم مشخص شده است.

متدولوژی EUP: در این متدولوژی نقش ها به طور کامل و جامع همراه با شرح وظایف و مصنوعاتی که در ساخت ان دخیل هستند مشخص شده است. در این متدولوژی ۲۵ نقش مختلف که عمدتا مربوط به فعالیت های سازمانی متدولوژی هستند به ۳۰ نقش موجود در RUP اضافه شده است و مجموعا ۵۵ نقش دارد.

مقایسه RUP با Fusion: در Fusion نقش ها به صراحت تعریف نشده اند و در این زمینه RUP که نقش ها در آن با دقت و شرح وظایف مشخص هستند برتری دارد.

مقایسه RUP با USDP: RUP نقش ها را با دقت بیشتر و جزئیات کامل تر از UP معرفی کرده است و این مورد هم با توجه به این است که UP کمی سطح بالا است.

مقایسه RUP با EUP: در این زمینه دو متدولوژی مشابه هستند و تعریف نقش ها و شرح وظایف و مصنوعات دخیل در هر دو دقیق تفاوت تعداد نقش های تعریفی است که به نوع متدولوژی ها بر می گردد.

زبان مدل سازی:

متدولوژی RUP: این متدلوژی از زبان UML برای مدل سازی استفاده می کند و استفاده از هر نوع زبان مدل سازی دیگری در آن ممنوع است.

متدولوژی Fusion: این متدولوژی دارای نشانه گذاری های جامع، ساده و به خوبی تعریف شده برای همه مدل های خود است و چون این نشانه گذاری ها بر اساس رویه های عملی موجود بوده یادگیری آنها آسان است.

متدولوژی USDP: زبان مدل سازی این متدولوژی UML است.

متدولوژی EUP: در این متدولوژی می توان از UML یا هر زبان مدل سازی که مناسب باشد استفاده کرد مثلا برای مدل سازی قلمرو کسب‌وکار می توان از DFD استفاده نمود.

مقایسه RUP با Fusion: زبان مدل سازی Fusion که مشابه به UML است ساده تر است ولی RUP که از UML استفاده می کند مدل های قدرتمندتری می سازد و می تواند همزمان مدل های صوری را هم تولید کند و در این زمینه بهتر از Fusion می باشد.

مقایسه RUP با USDP: در این زمینه دو متدولوژی مشابه هستند.

مقایسه RUP با EUP: هر دو می توانند از UML استفاده کنند که در آن صورت تفاوتی نمی کنند ولی EUP مزیتی دارد که زبان مدل سازی آن محدودیتی ندارد پس در صورتی که چیزی قدرتمند تر از UML بود هم می تواند از آن استفاده کند که برای همین آزاد گذاشتن دست استفاده کند نسبت به RUP مزیت دارد.

محصولات:

متدولوژی RUP: در این متدولوژی در هر فاز و در هر تکرار محصولات متنوعی تولید می شوند که خوبی توصیف شده اند. علاوه بر مصنوعاتی که در متدولوژی UP تولید می شود مصنوعات و مستندات تولید شده در این متدولوژی شامل این موارد است: مستند ارزیابی تکرار، مستند برنامه تکرار، مستند توجیه اقتصادی، مستند ترتیب کارها، مستند ارزیابی وضعیت، مستند سنجش پروژه، مستند برنامه ریزی پذیرش محصول، مستند برنامه ریزی مدیریت ریسک، مستند فهرست ریسک ها، مستند برنامه ریزی حل مشکلات، مستند برنامه ریزی سنجش، سند چشم انداز کسب‌وکار، مدل usecase کسب‌وکار، مدل تحلیل کسب‌وکار، مستند ارزیابی سازمان هدف، مستند قواعد کسب‌وکار، مستند مشخصات تکمیلی کسب‌وکار، واژه نامه کسب‌وکار،

سند معماری کسب‌وکار، مدل usecase، مستند برنامه ریزی یکپارچه سازی، مستند برنامه ریزی تست، مستند تحلیل تست، مستند برنامه ریزی مدیریت پیکربندی.

متدولوژی Fusion: محصولات هر فاز متدولوژی به خوبی مشخص شده اند. این متدولوژی شش محصول بصری (۳ تا در تحلیل و ۳ تا در طراحی) و سه محصول متنی (۲ تا در تحلیل و یکی در طراحی) تولید می کند. یک Object Model کلی که شامل سیستم و محیط است و قلمرو مسئله از لحاظ اشیا و ارتباط آنها ساخته می شود. یک Transaction Scenario ساخته می شود که نمایش ترتیب زمانی رویدادها در طول زمان می باشد. System Interface Diagram وجود دارد که رویدادهای بین سیستم و عامل بیرونی را با ترتیب زمانی نمایش می دهد و یکپارچه شده Transaction Scenario می باشد. lifecycle Model که نمایش توالی مجاز فراخوانی عملیات سیستم در طول عمر سیستم است ساخته می شود. در Operation Model جزئیات همه عملیات های سیستم که قبلا در System Interface Diagram و lifecycle Model نمایش داده شده اند را جمع آوری می کند. Object Iteration Graph برای توصیف عملیات سیستم که در Operation Model بودند استفاده می شود. Visibility Graph نمایش دهنده دید اشیا به هم و کلاینت و سرور بودن آنها نسبت به هم است. در Class Description اطلاعات تفصیلی در مورد کلاس مثل نام، سوپرکلاس بلافصل، ویژگی ها و متدهای آن توصیف می شود. در Inheritance Graph سلسله مراتبی کلاس هایی که در تحلیل شناسایی شده اند با فاکتور گیری ساختار های مشترک ایجاد می شود.

متدولوژی USDP: در این متدولوژی در هر فاز و در هر تکرار محصولات متنوعی تولید می شوند که خوبی توصیف شده اند. Usecase Diagram و جدول usecase برای مشخص کردن نیازمندی ها و جزئیات مربوط به آنها استفاده می شود. Class Diagram تحلیل مفاهیم کلیدی قلمرو کسب‌وکار را مدل می کند. Package Diagram ها کلاس ها را گروه بندی می کنند و سطح به سطح هم آنها را نمایش می دهند که می تواند نمایش معماری سیستم هم باشد. از Sequence Diagram برای نمایش تحقق Usecase ها استفاده می شود. قبل از ایجاد آن از Swimlane Activity برای تبدیل Usecase های غیر شی گرا به Sequence Diagram شی گرا استفاده می شود. همچنین برای نمایش جنبه های خاص و رفتارهای مهم Usecase ها هم می توان از Activity Diagram استفاده کرد که به دلیل ساختار ساده آن برای انواع ذینفعان قابل هم است. Class Diagram های طراحی ساخته می شوند که کامل و با جزئیات و آماده پیاده سازی هستند. از Component Diagram برای نمایش ارتباط اشیا و کلاس ها استفاده می شود و اینترفیس ها و زیر سیستم ها در آن مشخص می شود. در صورتی که پروژه بلادرنگ سخت یا تعبیه شده باشد می‌توان

با کمک Timing Diagram مدل سازی آن را انجام داد. از State Machine استفاده می شود تا رفتار پویای کلاس ها مدل شود. یک Deployment Diagram هم ساخته می شود که نمایش نحوه قرار گیری سیستم روی بستر و سخت افزار را نشان می دهد. همچنین مستندات و محصولاتی مثل سند چشم انداز، واژه نامه پروژه، برنامه پروژه، توجیه اقتصادی، ارزیابی ریسک، سند معماری، نمونه سازی دور ریختنی، راهنمای کاربری، توضیحات انتشارها و برنامه پشتیبانی کاربران هم تولید و ایجاد می شوند.

متدولوژی EUP: در این متدولوژی در هر فاز و در هر تکرار محصولات متنوعی تولید می شوند که خوبی توصیف شده اند. علاوه بر مصنوعاتی که در متدولوژی RUP تولید می شود مصنوعات و مستندات تولید شده در این متدولوژی شامل این موارد است: مدل معماری سازمانی، مستند نیازمندی های معماری کسب کار سازمانی، مدل فرایند کسب‌وکار سازمانی، مستند مشخصات قواعد کسب‌وکار سازمانی، مستند برنامه ریزی زیرساخت های محاسباتی سازمانی، راهنمای داده سازمانی، مدل داده سازمانی، راهنمای ایجاد سازمانی، مدل دامنه سازمانی، مستند راهنمای تسهیل گری سازمانی، مستند برنامه ریزی تسهیل گری سازمانی، مستند اهداف سازمانی، مالکیت معنوی سازمان، گواهی نامه های سازمانی، بیانیه ماموریت سازمان، مستند فهرست ریسک های سازمانی، مستند برنامه ریزی مدیریت ریسک های سازمانی، مستند راهنمای امنیت سازمانی، مستند برنامه ریزی امنیت سازمانی، چشم انداز سازمان، مستندات منابع داده موروثی، برنامه زیری موفقیت بلند مدت، ارزیابی نیازهای سازمان، مدل سازمانی، پورتفو، برنامه ریزی پورتفو، تعریف نقش ها، برنامه ریزی پیاده سازی فرایند، چشم انداز فرایند، راهنمای استفاده مجدد، برنامه ریزی سنجش استفاده مجدد، برنامه ریزی برنامه استفاده مجدد.

مقایسه RUP با Fusion: از نظر تعداد و غنی بودن مصنوعات و مستندات و پوشش در طول فازها RUP بسیار بهتر از Fusion است.

مقایسه RUP با USDP: RUP مصنوعات و مستندات UP را دارد و خودش هم به آنها اضافه کرده است و از لحاظ با پوشش دغدغه های بیشتر از UP بهتر است.

مقایسه RUP با EUP: EUP تمام مصنوعات RUP را دارد و با توجه به فازها و نظام هایی که اضافه نموده مصنوعات بیشتری نسبت به RUP دارد که از این لحاظ برتر است.

تکنیک ها و قواعد:

*متدولوژی **RUP***: تکنیک ها و رویه های عملی مناسب برای فعالیت های مختلف در این متدولوژی به خوبی بیان شده اند و در کتاب های مختلفی هم برای آن مقالات مختلف نگارش شده است. مثلا روش هایی برای نحوه مدیریت نیازمندی ها، مدیریت ریسک، تضمین کیفیت مداوم و معماری تایید شده در آن توضیح داده شده است.

*متدولوژی **Fusion***: این متدولوژی تکنیک ها و قواعدی را تعریف نموده است. مثلا دارای چک لیست برای تعیین پایان زمان فازها می باشد. همچنین دارای قواعدی برای نگاشت تحلیل به طراحی و طراحی به کد دارد. یا برای استخراج کلاس ها، ویژگی ها و متدهای آن استفاده از روش grammatical parsing را توصیه نموده است. همچنین تکنیک هایی برای سنجش و ردیابی عیوب در فرایند پروژه با جزئیات شرح داده شده است.

*متدولوژی **USDP***: تکنیک ها و رویه های عملی مناسب برای فعالیت های مختلف در این متدولوژی به خوبی بیان شده اند و در کتاب های مختلفی هم برای آن مقالات مختلف نگارش شده است. مثلا نحوه استخراج نیازمندی ها، مدل سازی usecase ها یا یافتن کلاس های طراحی در ان به خوبی توضیح داده شده اند.

*متدولوژی **EUP***: تکنیک ها و رویه های عملی مناسب برای فعالیت های مختلف در این متدولوژی به خوبی بیان شده اند و در کتاب های مختلفی هم برای آن مقالات مختلف نگارش شده است. در این متدولوژی رویه ها قواعدی برای مسائل سازمانی مطرح شده است.

*مقایسه **RUP** با **Fusion***: قواعد و تکنیک های RUP بسیار بیشتر و دقیق تر و کاربردی تر از Fusion برای اجرای پروژه هستند

*مقایسه **RUP** با **USDP***: در این زمینه دو متدولوژی مشابه هستند و RUP با توجه به گسترش تکنیک ها و قواعد بیشتری را نسبت بهUP دارد و این به دلیل کامل تر بودن RUP است.

*مقایسه **RUP** با **EUP***: این دو متدولوژی در این معیار مشابه هستند و EUP قواعد و تکنیک های مربوط به مدیریت سازمانی و برخی تکنیک های دیگر را بیشتر از RUP دارد.

فعالیت های چتری:

متدولوژی RUP: فعالیت های چتری این متدولوژی در سه نظام مدیریت پیکربندی و تغییرات، مدیریت پروژه و محیط ایجاد آن است که در طول فرایند گسترده هستند.

متدولوژی Fusion: این متدولوژی پوششی برای فعالیت های چتری به طور خاص ندارد.

متدولوژی USDP: تاکید روی فعالیت های چتری در این متدولوژی کمرنگ است و نظام خاص برای انجام آن ندارند و فعالیت ها در فازها انجام می شود.

متدولوژی EUP: فعالیت های چتری این متدولوژی در چهار نظام مدیریت پیکربندی و تغییرات، مدیریت پروژه و محیط ایجاد و مدیریت سازمانی آن است که البته خود مدیریت سازمانی شامل ۷ نظام می باشد که در طول فرایند گسترده هستند.

مقایسه RUP با Fusion: RUP با پوشش فعالیت های چتری بسیار بهتر از Fusion است که پوششی ندارد.

مقایسه RUP با USDP: UP فعالیت های چتری کمرنگی دارد و RUP بسیار بهتر از UP است و فعالیت های آن بهتر و پوشش بیشتری دارند و اینکه نظام برای ان ایجاد شده نشان از تاکید متدلوژی روی آن است.

مقایسه RUP با EUP: در این زمینه EUP با پوشش بیشتر فعالیت های چتری و داشتن دغدغه های سازمانی از RUP بهتر است.

معیار پوشش چرخه عمر عمومی ایجاد نرم‌افزار:

فعالیت های چرخه عمر عمومی شامل تعریف، ایجاد و نگهداری هستند که اگر این سه مرحله کامل پوشش داده شوند چرخه عمر کامل است.

تعریف:

کاوش قلمرو مسئله و مدل سازی آن:

متدولوژی RUP: در این متدولوژی در فاز آغاز قلمرو مسئله کاوش می شود و چیستی سیستم بدست می آید و چیزهایی که جز سیستم نیست هم مشخص می شوند. همچنین مدل سازی هم انجام می شود که Business Usecase Model، Business Object Model و Business Process Model را تولید می کند.

متدولوژی Fusion: ای متدلوژی قلمرو مسئله را به خوبی کاوش می کند و آن را توسط Overall Object Model که شامل سیستم و بیرون سیستم است مدل می کند. Transaction Scenario و System Interface Diagram هم مشخص کننده رویداد های ورودی و خروجی در مرز سیستم هستند.

متدولوژی USDP: در این متدولوژی در فاز آغاز قلمرو مسئله کاوش می شود و چیستی سیستم بدست می آید و چیزهایی که جز سیستم هم نیست هم مشخص می شوند. همچنین مدل سازی هم انجام می شود.

متدولوژی EUP: در این متدولوژی در فاز آغاز قلمرو مسئله کاوش می شود و چیستی سیستم بدست می آید و چیزهایی که جز سیستم هم نیست هم مشخص می شوند. همچنین مدل سازی هم انجام می شود که Business Usecase Model، Business Object Model و Business Process Model را تولید می کند.

مقایسه RUP با Fusion: در این زمینه هر دو قلمرو مسئله را کاوش می کنند ولی مدل ها و مستنداتی که RUP تولید می کند بسیار بهتر غنی تر می باشند.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند.

استخراج نیازمندی ها:

متدولوژی RUP: در این متدولوژی استخراج نیازمندی ها بروز کننده است. ابتدا در فاز آغاز ۱۰ تا ۲۰ درصد نیازمندی ها استخراج می شوند، در فاز تفصیل تا ۸۰ درصد استخراج می شود و در فایل ساخت هم باقیمانده نیازمندی های سیستم استخراج می شوند.

متدولوژی Fusion: استخراج نیازمندی ها ندارد و نیازمندی ها ورودی اصلی فرایند هستند.

متدولوژی USDP: در این متدولوژی استخراج نیازمندی ها بروز کننده است. ابتدا در فاز آغاز ۱۰ تا ۲۰ درصد نیازمندی ها استخراج می شوند، در فاز تفصیل تا ۸۰ درصد استخراج می شود و در فایل ساخت هم باقیمانده نیازمندی های سیستم استخراج می شوند.

متدولوژی EUP: در این متدولوژی استخراج نیازمندی ها بروز کننده است. ابتدا در فاز آغاز ۱۰ تا ۲۰ درصد نیازمندی ها استخراج می شوند، در فاز تفصیل تا ۸۰ درصد استخراج می شود و در فایل ساخت هم باقیمانده نیازمندی های سیستم استخراج می شوند.

مقایسه RUP با Fusion: متدولوژی Fusion فاز استخراج ندارد و نیازمندی ها ورودی آن هستند ولی RUP نیازمندی های وظیفه ای و غیر وظیفه ای را با روش هایی که توضیح می دهد استخراج می کند.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند.

امکان سنجی:

متدولوژی RUP: در فاز آغاز امکان سنجی انجام و بررسی می شود پروژه قابل انجام هست یا خیر. از نظر فنی می توان با نمونه سازی از معماری و از لحاظ اقتصادی با توجیه اقتصادی این امکان سنجی انجام می شود. در صورتی که نیاز باشد امکان سنجی های دیگر نیز انجام می شود و در انتهای این فاز یک go یا no-go برای انجام براساس امکان سنجی گفته می شود.

متدولوژی Fusion: این متدلوژی فعالیتی برای امکان سنجی پروژه ندارد.

متدولوژی USDP: در فاز آغاز امکان سنجی انجام و بررسی می شود پروژه قابل انجام هست یا خیر. از نظر فنی می توان با نمونه سازی از معماری و از لحاظ اقتصادی با توجیه اقتصادی این امکان سنجی انجام می شود. در صورتی که نیاز باشد امکان سنجی های دیگر نیز انجام می شود و در انتهای این فاز یک رفتن یا نرفتن برای انجام براساس امکان سنجی گفته می شود.

متدولوژی EUP: در فاز آغاز امکان سنجی انجام و بررسی می شود پروژه قابل انجام هست یا خیر. از نظر فنی می توان با نمونه سازی از معماری و از لحاظ اقتصادی با توجیه اقتصادی این امکان سنجی انجام می شود. در صورتی که نیاز باشد امکان سنجی های دیگر نیز انجام می شود و در انتهای این فاز یک go یا no-go برای انجام براساس امکان سنجی گفته می شود.

مقایسه RUP با Fusion: متدلوژی Fusion فعالیتی برای امکان سنجی ندارد و RUP با انجام امکان سنجی برتر از Fusion می باشد.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند.

ایجاد:

طراحی معماری:

متدولوژی RUP: طرح اولیه معماری در فاز آغاز بدست می آید و معماری مشخص می شود و در صورت نیاز نمونه دور ریختنی برای آن ساخته می شود. اما خود معماری در فاز تفصیل تکمیل و در انتهای این فاز هم معماری تثبیت شده سیستم باید ساخته شده باشد.

متدولوژی Fusion: این متدولوژی طراحی معماری را از نظر فیزیکی پوشش نمی دهد و طراحی معماری آن در منطقی است و این کار در فاز تحلیل متدولوژی انجام می شود. همچنین می توان کلاس ها را به کمک جدا سازی نمودار ها و به کمک قواعد موجود در متدولوژی که کمک به اشتراک گذاری کلاس ها و رابطه ها می دهد خوشه بندی کرد که نوعی معماری منطقی هم هست.

متدولوژی USDP: طرح اولیه معماری در فاز آغاز بدست می آید و معماری مشخص می شود و در صورت نیاز نمونه دور ریختنی برای آن ساخته می شود. اما خود معماری در فاز تفصیل تکمیل و در انتهای این فاز هم معماری تثبیت شده سیستم باید ساخته شده باشد.

متدولوژی EUP: طرح اولیه معماری در فاز آغاز بدست می آید و معماری مشخص می شود و در صورت نیاز نمونه دور ریختنی برای آن ساخته می شود. اما خود معماری در فاز تفصیل تکمیل و در انتهای این فاز هم معماری تثبیت شده سیستم باید ساخته شده باشد.

مقایسه RUP با Fusion: RUP معماری محور است و معماری را هم منطقی و هم فیزیکی می سازد اما Fusion در این حد به معماری توجه ندارد و RUP بهتر است.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند.

طراحی تفصیلی:

متدولوژی RUP: طراحی تفصیلی در این متدولوژی در نظام مربوط به تحلیل و طراحی آن انجام می شود که در کل فرایند پراکنده شده و میزان انجام آن بستگی به هدف فاز دارد اما بیشتر طراحی در فازهای تفصیل و ساخت انجام می شود.

*متدولوژی **Fusion***: طراحی تفصیلی در فاز طراحی متدولوژی انجام می شود که علاوه بر اضافه شدن کلاس های مربوط به پیاده سازی، مجموعه مصنوعات دیگری هم تولید می شوند.

*متدولوژی **USDP***: طراحی تفصیلی در این متدولوژی در نظام مربوط به طراحی آن انجام می شود که در کل فرایند پراکنده شده و میزان انجام آن بستگی به هدف فاز دارد اما بیشتر طراحی در فازهای تفصیل و ساخت انجام می شود.

*متدولوژی **EUP***: طراحی تفصیلی در این متدولوژی در نظام مربوط به تحلیل و طراحی آن انجام می شود که در کل فرایند پراکنده شده و میزان انجام آن بستگی به هدف فاز دارد اما بیشتر طراحی در فازهای تفصیل و ساخت انجام می شود.

*مقایسه **RUP** با **Fusion***: طراحی تفصیلی در RUP بسیار بهتر از Fusion است و مستندات و مصنوعات بیشتر و بهتری برای ان تولید می شود که مبنای پیاده سازی قرار بگیرد.

*مقایسه **RUP** با **USDP***: در این زمینه هر دو مشابه هستند.

*مقایسه **RUP** با **EUP***: در این زمینه هر دو مشابه هستند.

پیاده سازی:

*متدولوژی **RUP***: پیاده سازی مربوط به معماری در فاز تفصیل و پیاده سازی usecase ها در فاز ساخت انجام می شود. البته ممکن است usecase های با ریسک بالا در فاز ساخت هم پیاده سازی بشوند.

*متدولوژی **Fusion***: در این متدلوژی پیاده سازی سیستم در فاز پیاده سازی آن انجام می شود.

*متدولوژی **USDP***: پیاده سازی مربوط به معماری در فاز تفصیل و پیاده سازی usecase ها در فاز ساخت انجام می شود. البته ممکن است usecase های با ریسک بالا در فاز ساخت هم پیاده سازی بشوند.

*متدولوژی **EUP***: پیاده سازی مربوط به معماری در فاز تفصیل و پیاده سازی usecase ها در فاز ساخت انجام می شود. البته ممکن است usecase های با ریسک بالا در فاز ساخت هم پیاده سازی بشوند.

*مقایسه **RUP** با **Fusion***: در این زمینه هر دو پیاده سازی دارند و برتر ی بر یکدیگر ندارند و تفاوت آنها در محل های اجرای پیاده سازی و در Fusion در یک فاز است ولی در RUP حتی می توان در فاز آغاز هم پیاده سازی از نوع دور ریختنی داشت.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند.

تست:

متدولوژی RUP: تست ها در این متدولوژی در نظام تست آن انجام می شود و در تمام فازها پراکنده هستند. در فاز های تفصیل و ساخت تست های ریز دانه و مربوط به پیاده سازی ها انجام می شود و در فاز انتقال تست های درشت دانه و سیستمی و تست پذیرش انجام خواهد شد.

متدولوژی Fusion: در این متدولوژی برای انجام تست ها روی سیستم و تست های سیستمی فعالیتی معرفی نشده است و خود سازنده تست را به عنوان یک تکنیک بررسی ذکر کرده است و گفته است بهتر است همراه با کد نویسی انجام شود. اما Fusion Test Model را تست مدل های ساخته شده در تحلیل، Scenario Test را برای تست سناریو ها به گونه ای که در نیازمندی های کاربر تعیین شده که مشخص می کند سیستم چقدر خوب از سناریو های تعریف شده توسط کاربر در مدل ها پشتیبانی می کند. همچنین Object Interaction Test، Object Reference Test، Class Inheritance Test، Method Test، Object Test، Module Test هم تعریف شده اند. Application Test برای تست کامل نرم‌افزار و requirement Tracking Test برای تست رهگیری نیازمندی ها به کد و بالعکس تعریف شده اند. که این موارد بعدا به Fusion اضافه شده اند.

متدولوژی USDP: تست ها در این متدولوژی در نظام تست آن انجام می شود و در تمام فازها پراکنده هستند. در فاز های تفصیل و ساخت تست های ریز دانه و مربوط به پیاده سازی ها انجام می شود و در فاز انتقال تست های درشت دانه و سیستمی و تست پذیرش انجام خواهد شد.

متدولوژی EUP: تست ها در این متدولوژی در نظام تست آن انجام می شود و در تمام فازها پراکنده هستند. در فاز های تفصیل و ساخت تست های ریز دانه و مربوط به پیاده سازی ها انجام می شود و در فاز انتقال تست های درشت دانه و سیستمی و تست پذیرش انجام خواهد شد.

مقایسه RUP با Fusion: در RUP تاکید روی تست بسیار بیشتر است در Fusion فعالیت مخصوص تست وجود ندارد و به نوعی توسط طراحان بودن تست یک پیشفرض قرار گرفته است و در این زمینه RUP بسیار بهتر است و انواع تست ها را انجام می دهد.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند.

استقرار:

متدولوژی RUP: در این متدولوژی استقرار در طول نظام استقرار در طی فازهای ساخت و انتقال انجام می شود و در آنجا نرم‌افزار در محیط کاربر استقرار پیدا می کند.

متدولوژی Fusion: در این متدولوژی فعالیتی برای استقرار نرم‌افزار تعریف نشده است

متدولوژی USDP: این متدولوژی نظام خاص برای استقرار ندارد و فعالیت استقرار در فازهای های به صورت کمرنگ وجود دارد که در قالب نظام پیاده سازی انجام می پذیرد.

متدولوژی EUP: در این متدولوژی استقرار در طول نظام استقرار در طی فازهای ساخت و انتقال انجام می شود و در آنجا نرم‌افزار در محیط کاربر استقرار پیدا می کند.

مقایسه RUP با Fusion: متدولوژی RUP با داشتن فرایند استقرار و تعریف ان بهتر از Fusion می باشد.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند.

نگهداری:

متدولوژی RUP: این متدولوژی فاز نگهداری را ندارد و گفته برای نگهداری یکبار چرخه اجرا شود که عملا برای نگهداری نامناسب است.

متدولوژی Fusion: در این متدولوژی فعالیتی برای نگهداری نرم‌افزار تعریف نشده است

متدولوژی USDP: این متدولوژی فاز نگهداری را ندارد و گفته برای نگهداری یکبار چرخه اجرا شود که عملا برای نگهداری نامناسب است.

متدولوژی EUP: در این متدولوژی با اضافه کردن نظام عملیات و پشتیبانی گفته شده که نگهداری دارد ولی عملا نگهداری را پوش نمی دهد.

مقایسه RUP با Fusion: متدلوژی RUP گفته است برای نگهداری فرایند را دوباره اجرا کنید و اشاره به آن کرده ولی Fusion صحبتی در این مورد نکرده است و RUP در اینجا بهتر است.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: متدلوژی EUP نظام عملیات و پشتیبانی را برای نگهداری اضافه کرده ولی عملا نگهداری ندارد و فقط پشتیبانی و عملیات است. ولی در این زمینه بهتر از RUP است که حتی برای این کار نظام خاص تعریف نکرده است.

معیار پشتیبانی از فعالیت های چتری:

متدولوژی باید فعالیت های چتری را پوشش بدهد. فعالیت های چتری فعالیت هایی هستند که در همه بخش های چرخه فرایند لحاظ می شوند و شامل مدیریت ریسک از طریق ارزیابی ریسک و فعالیت های کاهش ریسک گنجانده شده در چرخه عمر، مدیریت پروژه از طریق برنامه ریزی، زمان بندی و تکنیک های کنترلی قرار گرفته در فرایند و تضمین کیفیت با ارزیابی کیفیت و تکنیک های بهبود کیفیت در فرایند هستند.

مدیریت ریسک:

متدولوژی RUP: این متدولوژی بر مبنای ریسک کار می کند و ریسک ها در فاز آغاز شناسایی و در مستند ارزیابی ریسک ثبت و اولویت بندی می شوند. ریسک های مهم باید در فاز آغاز رفع شوند که ممکن است ریسک های فنی با کمک نمونه سازی دور ریختنی رفع شود. همچنین در صورتی که ریسکی نباشد که امکان پذیری پروژه را دچار مشکل کند به فاز بعد می رود. در انتهای فاز تفصیل باید همه ریسک ها رفع شده باشند و نباید ریسکی باقیمانده باشد. در این فاز حتی ممکن است usecase های پر ریسک روی معماری تثبیت شده پیاده سازی شوند. همچنین تکراری افزایشی بودن متدولوژی هم باعث مدیریت ریسک می شود.

متدولوژی Fusion: در این متدولوژی فعالیتی برای مدیریت ریسک وجود ندارد.

متدولوژی USDP: این متدولوژی بر مبنای ریسک کار می کند و ریسک ها در فاز آغاز شناسایی و در مستند ارزیابی ریسک ثبت و اولویت بندی می شوند. ریسک های مهم باید در فاز آغاز رفع شوند که ممکن است ریسک های فنی با کمک مونه سازی دور ریختنی رفع شود. همچنین در صورتی که ریسکی نباشد که امکان پذیری پروژه را دچار مشکل کند به فاز بعد می رود. در انتهای فاز تفصیل باید همه ریسک ها رفع شده باشند و نباید ریسکی باقیمانده باشد. در این فاز ممکن است حتی usecase های پر ریسک هم روی معماری تثبیت شده پیاده سازی شوند. همچنین تکراری افزایشی بودن متدولوژی هم باعث مدیریت ریسک می شود.

متدولوژی EUP: این متدولوژی بر مبنای ریسک کار می کند و ریسک ها در فاز آغاز شناسایی و در مستند ارزیابی ریسک ثبت و اولویت بندی می شوند. ریسک های مهم باید در فاز آغاز رفع شوند که ممکن است

ریسک های فنی با کمک مونه سازی دور ریختنی رفع شود. همچنین در صورتی که ریسکی نباشد که امکان پذیری پروژه را دچار مشکل کند به فاز بعد می رود. در انتهای فاز تفصیل باید همه ریسک ها رفع شده باشند و نباید ریسکی باقیمانده باشد. در این فاز ممکن است حتی usecase های پر ریسک هم روی معماری تثبیت شده پیاده سازی شوند. همچنین تکراری افزایشی بودن متدولوژی هم باعث مدیریت ریسک می شود. مدیریت ریسک سازمانی هم دارد که شامل مدیریت ریسک هم در سطح برنامه و هم پورتفو است.

مقایسه RUP با Fusion: در RUP مدیریت ریسک وجود دارد و در Fusion فعالیتی برای آن نیست پس RUP بهتر عمل می کند.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: متدلوژی EUP تمام فرایندهای مربوط به مدیریت ریسک در RUP را دارد به علاوه فعالیت هایی هم برای مدیریت ریسک در سطح سازمان دارد که از این لحاظ بر RUP برتری دارد.

مدیریت پروژه:

متدولوژی RUP: در این متدولوژی فعالیت های مدیریت پروژه به طور خاص در نظام مدیریت پروژه انجام می شود که شامل فعالیت های مربوط به زمان بندی پروژه، فازها و تکرارها و مدیریت تیم می باشد.

متدولوژی Fusion: در این متدولوژی فعالیتی برای مدیریت پروژه نیست. اما بعد ها به آن مواردی اضافه شد که با بررسی زمان های صرف شده، تعداد کلاس ها، تعداد روابط آنها و تعداد عملیات می توانستند فازهای پروژه را تخمین بزنند و زمان بندی کنند. همچنین با جمع آوری زمان ها و ثبت آن در Progress Flow Chart برای فازهای مختلف مدیر پروژه می توانست برای آینده برنامه ریزی کند.

متدولوژی USDP: برای مدیریت پروژه در این متدولوژی نظام خاصی وجود ندارد و فعالیت های مدیریتی مربوط به زمان بندی ها در خود فازها انجام می پذیرند. در مورد مدیریت تیم ها متدولوژی اشاره صریحی نداشته است.

متدولوژی EUP: در این متدولوژی فعالیت های مدیریتی تحت نظام مدیریت پروژه انجام می شود و زمان بندی فازها، تکرارها و پروژه در ان انجام می شود. به علاوه یک نظام مدیریت سازمانی هم دارد که با این نظام فعالیت های مدیریتی هم در سطح پروژه بسیار قوی انجام می شود.

مقایسه RUP با Fusion: در RUP مدیریت پروژه تعریف شده و در نظام خاصی فعالیت های مربوطه را انجام می دهد و از این لحاظ از Fusion که فعالیتی برای آن تعریف نکرده است بهتر است.

مقایسه RUP با USDP: در UP فعالیت های مدیریت پروژه در فازها به طور ضمنی وجود دارد و خیلی هم کمرنگ هستند در حالی که RUP دارای نظام خاص برای مدیریت پروژه می باشد و از این لحاظ برتر از UP است.

مقایسه RUP با EUP: متدلوژی EUP علاوه بر اینکه تمام فعالیت های مدیریت پروژه RUP را دارد فرایند های مدیریتی در سطح سازمان را هم پوشش می دهد و از این نظر بهتر از RUP می باشد.

تضمین کیفیت:

متدولوژی RUP: برای تضمین کیفیت در این متدولوژی همراه با پیاده سازی باید کد تست شود. همچنین در فاز انتقال هم تست های کامل سیستمی انجام می شود که می تواند به افزایش کیفیت کمک کند. همچنین معیارهایی را هم برای افزایش کیفیت کد تعریف کرده است. انجام تست های تکراری به جای تست های یکباره روش دیگری برای تضمین کیفیت مداوم سیستم است. همچنین مبتنی بر نیازمندی ها بودن می تواند به تضمین کیفیت کمک نماید

متدولوژی Fusion: در این متدولوژی به صراحت درباره تضمین کیفیت صحبت نشده اما چک لیست هایی که تکمیل آنها مشخص کننده اتمام فاز ها هستند را می توان به نوعی تضمین کیفیت دانست. همچنین Fusion Test Model را معرفی کرده که با تست مدل ها تعداد عیوب را کمینه کردن و کیفیت محصول تولید شده را بهینه می کند. فعالیت های مربوط به دقیق کردم کلاس ها و فرایند های مربوط به ارث بری که باعث Maintainable شدن کد می شود را هم می توان نوعی فرایند تضمین کیفیت دانست.

متدولوژی USDP: برای تضمین کیفیت در این متدولوژی همراه با پیاده سازی باید کد تست شود. همچنین در فاز انتقال هم تست های کامل سیستمی انجام می شود که می تواند به افزایش کیفیت کمک کند. انجام تست های تکراری به جای تست های یکباره روش دیگری برای تضمین کیفیت مداوم سیستم است. همچنین استفاده از TDD را برای تضمین کیفیت مورد توجه قرار داده است. همچنین مبتنی بر نیازمندی ها بودن می تواند به تضمین کیفیت کمک نماید

متدولوژی EUP: برای تضمین کیفیت در این متدولوژی همراه با پیاده سازی باید کد تست شود. همچنین در فاز انتقال هم تست های کامل سیستمی انجام می شود که می تواند به افزایش کیفیت کمک کند. انجام تست های تکراری به جای تست های یکباره روش دیگری برای تضمین کیفیت مداوم سیستم است. . همچنین استفاده از TDD را برای تضمین کیفیت مورد توجه قرار داده است. در مورد محصولات قابل استفاده مجدد در مخزن سازمان برای هر کدام یک مستند تضمین کیفیت هم تولید می کند.. همچنین مبتنی بر نیازمندی ها بودن می تواند به تضمین کیفیت کمک نماید

مقایسه RUP با Fusion: در RUP فعالیت های بیشتری برای تضمین کیفیت وجود دارد. در Fusion برای تضمین کیفیت مدل ها چک لیست هایی وجود دارد. همچنین در آن به تست اشاره نشده ولی آن را دارد و به نظر می رسد سازندگان آن را به عنوان پیشفرض در نظر گرفته اند. در مقایسه این دو RUP بهتر از Fusion است.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در EUP علاوه بر فرایند های تضمین کیفیت RUP فرایند هایی برای تضمین کیفیت در سطح سازمان وجود دارد که از نظر بهتر از RUP است.

معیار بی‌درزی و گذر هموار:

بی درزی به معنی عدم وجود تغییر الگو در زنجیره ی مدل سازی و فعالیت ها است. گذار بین فازها، مراحل و تکالیف باید هموار باشند و باید ادامه طبیعی هم باشند و میان آنها وقفه ایجاد نشود.

بی درزی و گذار هموار:

متدولوژی RUP: در این متدولوژی به دلیل بر مبنای نیازمندی ها بودن درز وجود ندارد البته به دلیل ذات شی گرا به طور طبیعی در تبدیل usecase های غیر شی گرا به sequence diagram شی گرا دارای درز است که آن را با کمک swimlane Activity Diagram در تبدیل usecase ها به sequence diagram تا حدی رفع می کنند. همچنین استفاده از UML و یکپارچه بودن آن و تکمیل مدل ها به مرور باعث می شود که گذر ها هموار باشد.

متدولوژی Fusion: این متدولوژی در مجموع بی درز است اما تبدیل نیازمندی ها به مدل شی گرا به صورت ذاتی دارای درز است. همچنین گذار از task به task و phase به phase هموار است و دقیقا مشخص است چه چیزی به چه چیزی نگاشت می شود. با کمک ابزار Paradigm Plus می توان با اسکریپت های

سفارشی شده انتقال بین فازها را خودکاری سازی تولید مدل ها و پیشنهاد شکستن آنها خودکار کرد که این هم می تواند باعث هموار شدن کامل گذر بین مدل ها شود.

متدولوژی USDP: در این متدولوژی به دلیل بر مبنای نیازمندی ها بودن درز وجود ندارد البته به دلیل ذات شی گرا به طور طبیعی در تبدیل usecase های غیر شی گرا به sequence diagram شی گرا دارای درز است که آن را با کمک swimlane Activity Diagram در تبدیل usecase ها به sequence diagram تا حدی رفع می کنند. همچنین استفاده از UML و یکپارچه بودن آن و تکمیل مدل ها به مرور باعث می شود که گذر ها هموار باشد.

متدولوژی EUP: در این متدولوژی به دلیل بر مبنای نیازمندی ها بودن درز وجود ندارد البته به دلیل ذات شی گرا به طور طبیعی در تبدیل usecase های غیر شی گرا به sequence diagram شی گرا دارای درز است که آن را با کمک swimlane Activity Diagram در تبدیل usecase ها به sequence diagram تا حدی رفع می کنند. اما چون متدولوژی استفاده از UML را اجبار کرده ممکن است مدل هایی ساخته شود که به بی درزی و گذر هموار آسیب بزند. مثلا استفاده از DFD برای مدل سازی قلمرو کسب‌وکار می تواند باعث آسیب به بی درزی شود که البته سازندگان متدولوژی گفته اند مزیت های به داشتن درز در متدولوژی می ارزد.

مقایسه RUP با Fusion: از نظر بی درزی هر دو مشابه هستند ولی اینکه RUP روشی برای تبدیل مدل شی گرا به غیر شی گرا دارد باعث برتری آن است. در مورد گذر هموار تعداد زیاد مدل ها می تواند به هموار تر بودن ان نسبت به Fusion آسیب بزند اما با کمک UML خیلی خوب می توان گذر هموار را مدیریت کرد و از این لحاظ هم بهتر از Fusion است.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: هم از نظر بی درزی و م گذر هموار RUP بهتر است. اولا چون EUP می تواند مدل هایی تولید کند که بی درزی را از بین ببرد. همچنین تعدد مدل های آن و عدم اجبار به استفاده از UML می تواند باعث شود هموار شدن گذرها سخت تر از RUP باشد.

معیار مبتنی بودن بر نیازمندی ها:

کارها باید مبتنی بر نیازمندی ها باشند. نیازمندی های وظیفه ای و غیر وظیفه ای باید در اوایل فرایند متدولوژی استخراج شوند، در جای درست خود مدل سازی شده و مبنای کارهای طراحی، پیاده سازی و تست قرار گیرند.

متدولوژی RUP: در این متدولوژی همه چیز بر مبنای نیازمندی ها است که در قالب usecase بیان می شوند و مبنای همه چیز قرار می گیرند. همچین نیازمندی های غیرعملکردی هم استخراج می شوند که عمدتا محدودیت و ویژگی های کیفی هستند که البته در قالب usecase نمایش داده نمی شوند.

متدولوژی Fusion: در این متدولوژی با اینکه فاز استخراج نیازمندی ها را ندارد اما سند ورودی نیازمندی ها مبنای تمام فعالیت های بعدی قرار می گیرد و نیازمندی ها در فاز تحلیل به operation ها نگاشت می شوند. اما چون نیازمندی ها جداگانه مدل نمی شوند نمی توان گفت به طور مستقیم مبتنی بر نیازمندی ها است. در این متدولوژی صحبتی در مورد نیازمندی های غیرعملکردی وجود ندارد

متدولوژی USDP: در این متدولوژی همه چیز بر مبنای نیازمندی ها است که در قالب usecase بیان می شوند و مبنای همه چیز قرار می گیرند. همچین نیازمندی های غیرعملکردی هم استخراج می شوند که عمدتا محدودیت و ویژگی های کیفی هستند که البته در قالب usecase نمایش داده نمی شوند.

متدولوژی EUP: این متدولوژی بر مبنای نیازمندی ها است ولی نحوه مدل سازی نیازمندی ها را بر عهده استفاده کنندگان گذاشته و روش خاصی را برای مدل سازی اجبار نکرده است. همچین نیازمندی های غیرعملکردی هم استخراج می شوند که عمدتا محدودیت و ویژگی های کیفی هستند.

مقایسه RUP با Fusion: در Fusion به دلیل نداشتن فاز استخراج نیازمندی ها مبتنی بر نیازمندی ها به صورت مستقیم نیست و RUP با داشتن استخراج نیازمندی ها بهتر از Fusion است.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند و تفاوت در نحوه مدل سازی نیازمندی ها است که اثری در مبتنی بودن بر نیازمندی های آنها ندارد.

معیار آزمون پذیری، ملموس بودن و قابل رهگیری بودن به نیازمندی ها:

مصنوعات باید آزمون پذیر، ملموس و قابل فهم و قابل نگاشت به نیازمندی ها باشند.

آزمون پذیری:

متدولوژی RUP: این متدولوژی دارای مدل ها و مصنوعات زیاد است و پیچیدگی و وابستگی میان آنها باعث می شود قابلیت آزمون پذیری متدولوژی به خصوص در صورت تغییر در مدل ها کم باشد. مبتنی بر usecase بودن باعث می شود قابلیت تست تا حدی کاهش یابد.

متدولوژی Fusion: در این متدولوژی تعداد محصولات زیاد است ولی قابل فهم هستند و می توان تا حدودی با چک لیست های اتمام فازها آنها را تست کرد.

متدولوژی USDP: این متدولوژی دارای مدل ها و مصنوعات زیاد است و پیچیدگی و وابستگی باعث می شود قابلیت آزمون پذیری متدولوژی به خصوص در صورت تغییر در مدل ها کم باشد. مبتنی بر usecase بودن باعث می شود قابلیت تست تا حدی کاهش یابد.

متدولوژی EUP: این متدولوژی دارای مدل ها و مصنوعات بسیار زیاد است و پیچیدگی باعث می شود قابلیت آزمون پذیری متدولوژی به خصوص در صورت تغییر در مدل ها کم باشد. مبتنی بر usecase بودن باعث می شود قابلیت تست تا حدی کاهش یابد.

مقایسه RUP با Fusion: در Fusion محصولات کمتر و پیچیدگی آنها هم کمتر است پس بهتر از RUP می توان آن را تست کرد.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند و UP تنها به دلیل کمتر بودن مدل هایش برتری نامحسوسی دارد.

مقایسه RUP با EUP: اگر EUP از UML استفاده نکند مدیریت پیچیدگی هایش سخت می شود بنابراین در این زمینه RUP بهتر است. البته تعداد مدل های بیشتر در EUP می تواند قابلیت آزمون پذیری آن را نسبت به RUP کاهش دهد.

ملموس بودن:

متدولوژی RUP: در این متدولوژی در صورتی برای پروژه های مناسبش استفاده شود مصنوعات ملموس هستند ولی در صورتی که مثلا برای پروژه های کوچک استفاده شود برخی از مصنوعات آن ملموس نیستند و مشخص نیست به چه دردی می خورند.

*متدولوژی **Fusion***: در این متدولوژی با وجود دارای بودن تعداد زیادی مصنوع اما همگی قابل فهم هستند و علت وجودی آنها برای برنامه نویسان که استفاده کننده نهایی از آنها هستند کاملا واضح است.

*متدولوژی **USDP***: در این متدولوژی در صورتی برای پروژه های مناسبش استفاده شود مصنوعات ملموس هستند ولی در صورتی که مثلا برای پروژه های کوچک استفاده شود برخی از مصنوعات آن ملموس نیستند و مشخص نیست به چه دردی می خورند.

*متدولوژی **EUP***: در این متدولوژی در صورتی برای پروژه های مناسبش استفاده شود مصنوعات ملموس هستند ولی در صورتی که مثلا برای پروژه های کوچک یا پروژه های غیر سازمانی استفاده شود برخی از مصنوعات آن ملموس نیستند و مشخص نیست به چه دردی می خورند.

*مقایسه **RUP** با **Fusion***: در Fusion محصولات ملموس تر هستند ولی RUP به دلیل محصولات زیاد و پیچیدگی های بالا ممکن است علت وجودی محصول مشخص نباشد و این می تواند با توجه به سایز و بحرانیت پروژه ها تغییر کند.

*مقایسه **RUP** با **USDP***: در این زمینه هر دو مشابه هستند.

*مقایسه **RUP** با **EUP***: در این زمینه هر دو مشابه هستند و اینکه محصولات کدام یک ملموس تر است بستگی به پروژه ای برای آن استفاده می شود دارد.

قابل رهگیری به نیازمندی ها:

*متدولوژی **RUP***: این متدولوژی بر مبنای نیازمندی ها است که به usecase مدل می شوند و در ادامه همه چیز بر مبنای نیازمندی ها است و می توان نیازمندی ها را رهگیری کرد که خود UML هم به این کار کمک زیادی می کند. برای رهگیری نیازمندی ها از ماتریس رهگیری و ابزارRose هم می تواند استفاده کند.

*متدولوژی **Fusion***: در این متدولوژی رهگیری به نیازمندی ها توسط سناریو های سیستم انجام می شود و تمام مدل ها باید transaction scenario ها را محقق کنند و اینها مبنای تست هم قرار می گیرند.

*متدولوژی **USDP***: این متدولوژی بر مبنای نیازمندی ها است که به usecase مدل می شوند و در ادامه همه چیز بر مبنای نیازمندی ها است و می توان نیازمندی ها را رهگیری کرد که خود UML هم به این کار کمک زیادی می کند. برای رهگیری نیازمندی ها از ماتریس رهگیری هم استفاده می کند.

متدولوژی EUP: این متدولوژی بر مبنای نیازمندی ها در ادامه همه چیز بر مبنای نیازمندی ها است و اما عدم استفاده از UML و پیچیدگی بالای آن ممکن است باعث مشکل شدن رهگیری نیازمندی ها و نیاز به تلاش بیشتر برای آن باشد. برای رهگیری نیازمندی ها از ماتریس رهگیری هم استفاده می کند.

مقایسه RUP با Fusion: در RUP با توجه به اینکه مبتنی بر نیازمندی ها است و از ماتریس رهگیری استفاده می کند رهگیری محصولات بهتر از Fusion است.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه هر دو مشابه هستند.

معیار تشویق به مشارکت فعالانه کاربران:

کاربران باید به طور فعالانه در فرایند ایجاد نرم‌افزار دخالت داشته باشند که این کار در مدیریت مخاطرات و تضمین کیفیت اهمیت دارد.

متدولوژی RUP: در این متدولوژی مشارکت کاربران در استخراج نیازمندی ها وجود دارد همچنین می توانند در پیاده سازی نیازمندی های پر ریسک می توانند تاییدیه بدهند. در انتهای کار پیاده سازی نیز باید توسط کاربران تایید شود. البته مانند متدولوژی های چابک عضو در تیم ندارند. تست های پذیرش نوع دیگر حضور کاربران است. به علاوه تست های بتا هم شامل حضور فعال کاربران است.

متدولوژی Fusion: در مورد حضور فعال کاربران در مراحل ایجاد نرم‌افزار در این متدولوژی صحبتی نشده است و صرفا حضور کاربران در ابتدای مدل سازی و مصاحبه با آنها برای شناخت قلمرو مسئله است.

متدولوژی USDP: در این متدولوژی مشارکت کاربران در استخراج نیازمندی ها وجود دارد همچنین می توانند در پیاده سازی نیازمندی های پر ریسک می توانند تاییدیه بدهند. در انتهای کار پیاده سازی نیز باید توسط کاربران تایید شود. البته مانند متدولوژی های چابک عضو در تیم ندارند. تست های پذیرش نوع دیگر حضور کاربران است. به علاوه تست های بتا هم شامل حضور فعال کاربران است.

متدولوژی EUP: در این متدولوژی مشارکت کاربران در استخراج نیازمندی ها وجود دارد که از طریق کارگاه، مصاحبه و پرسش نامه است همچنین می توانند در پیاده سازی نیازمندی های پر ریسک می توانند تاییدیه بدهند. در انتهای کار پیاده سازی نیز باید توسط کاربران تایید شود. البته مانند متدولوژی های چابک عضو در تیم ندارند. تست های پذیرش نوع دیگر حضور کاربران است. به علاوه تست های بتا هم شامل حضور فعال کاربران است. همچنین فرایند هایی برای مرور و بازبینی در سطح سازمان هم دارد.

مقایسه RUP با Fusion: در این زمینه RUP فعالیت های بیشتری دارد که کاربران در ان دخیل هستند و از این لحاظ از Fusion بهتر است.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در EUP با توجه با داشتن برخی فرایند های سازمانی موارد بیشتری از حضور کاربران در آن وجود دارد و از لحاظ بهتر از RUP است.

معیار قابل اجرا بودن و کارا بودن:

باید بتوان فرایند را به راحتی اجرا کرد و در ایجاد نرم‌افزار به کار گرفت همچنین باید بتوان از فرایند به طور کارا، موثر و مفید در حوزه کاربرد استفاده کرد.

قابل اجرا بودن:

متدولوژی RUP: قابل اجرا بودن آن کاملا بستگی به نوع پروژه آن دارد به دلیل پیچیدگی های بالا و تعداد مدل های زیاد ممکن است برای برخی پروژه ها قابل اجرا باشد و برای برخی نباشد.

متدولوژی Fusion: این متدولوژی به دلیل سادگی و قابل فهم بودن محصولات به خوبی قابل اجراست.

متدولوژی USDP: قابل اجرا بودن آن کاملا بستگی به نوع پروژه آن دارد به دلیل پیچیدگی های بالا و تعداد مدل های زیاد ممکن است برای برخی پروژه ها قابل اجرا باشد و برای برخی نباشد.

متدولوژی EUP: قابل اجرا بودن آن کاملا بستگی به نوع پروژه آن دارد به دلیل پیچیدگی های بسیار بالا و تعداد مدل های زیاد ممکن است برای برخی پروژه ها قابل اجرا باشد و برای برخی نباشد.

مقایسه RUP با Fusion: در RUP سطحی از فرمالیزم وجود دارد و علاوه بر این معماری محور است و با وجود اینکه مدل های بسیار زیاد و پیچیده ای دارد قابلیت اجرای آن گستره بیشتری از Fusion دارد.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند.

مقایسه RUP با EUP: در این زمینه بسته به نوع پروژه و سطوح بحرانیت قابلیت اجرا متفاوت است. برای پروژه های سازمانی EUP با توجه به توجه به فعالیت های سازمانی قابل اجرا تر است. اما چون EUP به طور قطعی فرمالیزم ندارد در این زمینه RUP به دلیل داشتن فرمالیزم می تواند برای پروژه های با بحرانیت بالاتر هم استفاده بشود.

کارا بودن:

متدولوژی RUP: اجرای کارا بستگی به نوع پروژه و ابزارهای به کار گرفته شده دارد. اگر متدولوژی به خوبی سفارشی سازی شود که البته به دلیل پیچیدگی مشکل است و از ابزارهای مناسب برای مدیریت پیچیدگی های آن استفاده شود متدولوژی در پروژه مناسب خود کارا است. برای اجرای کارا استفاده از ابزار های شرکت Rational و دیگر ابزارها پیشنهاد شده که این می تواند اجرای کارا را به دلیل وابستگی به ابزار کاهش بدهد.

متدولوژی Fusion: این متدلوژی را برای سیستم هایی که دارای کلاس هایی زیادی نباشند می توان به صورت کارا اجرا کرد و برای سیستم های بزرگ به دلیل تعدد کلاس ها ممکن است کارای افت کند. البته ابزار هایی وجود دارد که بتواند باعث اجرای کارای متدولوژی بشود. مثلا برای رسم کارای مدل های می توان از Visio استفاده نمود. همچنین با کمک ابزار FusionCaseld می توان ابزارهای کنترلی تولید کد، مدل سازی کامل مبتنی بر توارث و ابزارهای تولید مستندات داشت که بر کارای می تواند اثری منفی داشته باشد.

متدولوژی USDP: اجرای کارا بستگی به نوع پروژه و ابزارهای به کار گرفته شده دارد. اگر متدولوژی به خوبی سفارشی سازی شود و از ابزارهای مناسب برای مدیریت پیچیدگی های آن استفاده شود متدولوژی در پروژه مناسب خود کارا است. این متدولوژی هم برای اجرای کارا ابزارهایی معرفی کرده اما به دلیل کوچک بودن متدولوژی کمتر می تواند روی کارایی اثر بگذارد.

متدولوژی EUP: اجرای کارا بستگی به نوع پروژه و ابزارهای به کار گرفته شده دارد. اگر متدولوژی به خوبی سفارشی سازی شود که البته به دلیل پیچیدگی بالا بسیار مشکل است و از ابزارهای مناسب برای مدیریت پیچیدگی های آن استفاده شود متدولوژی در پروژه مناسب خود کارا است. برای اجرای کارا استفاده از ابزار های شرکت Rational و دیگر ابزارها پیشنهاد شده و به دلیل گسترده بودن سطح استفاده ز ابزارهای آن بسیار بیشتر است که این می تواند اجرای کارا را به دلیل وابستگی به ابزار کاهش بدهد.

مقایسه RUP با Fusion: به دلیل پیچیدگی کمتر و وابستگی کمتر به ابزار می توان Fusion را کاراتر از RUP اجرا نمود. البته نداشتن معماری در Fusion برای آن ضعف محسوب می شود و اجرای کارا را کاهش می دهد که در این زمینه RUP چون معماری را طراحی می کند و معماری محور است بهتر از Fusion است.

مقایسه RUP با USDP: در این زمینه هر دو مشابه هستند اما UP به دلیل وابستگی کمتر به ابزار بهتر است.

مقایسه RUP با EUP: در این زمینه هر دو تا حدی مشابه هستند اما گسترده تر بودن مدل ها و پیچیدگی ها باعث می شود EUP وابسته تر به ابزار باشد و در این مورد RUP بهتر است.

معیار قابل مدیریت بودن پیچیدگی:

فرایند از تعدادی واحد کاری، نقش، محصول و تعدادی معیار سنجش تشکیل می شود. باید خود این مجموعه از نظر پیچیدگی قابل مدیریت باشد. مدیریت معمولا به دو روش بخش بندی و لایه بندی انجام می شود.

متدولوژی RUP: در این متدولوژی اجرای فاز به فاز و هر فاز به صورت تکراری افزایشی کمک به مدیریت پیچیدگی ها می کند. که فاز ها بخش بندی و تکرار ها لایه بندی هستند.

متدولوژی Fusion: این متدولوژی به غیر از اجرای کارها در سه فاز و صورت ترتیبی که مدیریت پیچیدگی به کمک بخش بندی است روش دیگری ندارد.

متدولوژی USDP: در این متدولوژی اجرای فاز به فاز و هر فاز به صورت تکراری افزایشی کمک به مدیریت پیچیدگی ها می کند. که فاز ها بخش بندی و تکرار ها لایه بندی هستند.

متدولوژی EUP: در این متدولوژی اجرای فاز به فاز و هر فاز به صورت تکراری افزایشی کمک به مدیریت پیچیدگی ها می کند. که فاز ها بخش بندی و تکرار ها لایه بندی هستند.

مقایسه RUP با Fusion: در این زمینه با توجه به اینکه RUP پروژه را به موارد ریزتری می شکند راحت تر می توان آنها را مدیریت کرد و بهتر از Fusion می باشد.

مقایسه RUP با USDP: در این زمینه تا حدی مشابه هستند اما RUP چون نظام های بیشتری دارد کارها را ریزتر می کند و این می تواند به مدیریت پیچیدگی ها کمک بیشتری کند و از این لحاظ بهتر است.

مقایسه RUP با EUP: در این زمینه تا حدی مشابه هستند اما EUP چون نظام های بیشتری دارد کارها را ریزتر می کند و این می تواند به مدیریت پیچیدگی ها کمک بیشتری کند و از این لحاظ بهتر است.

معیار قابلیت گسترش، مقیاس پذیری، پیکربندی و انعطاف پذیری:

با بتوان فرایند یک متدولوژی را با کمک نقاط گسترش و مکانیزم های به صراحت تعریف شده آن گسترش داد، در پروژه های با اندازه مختلف و سطوح بحرانیت مختلف اعمال کرد، در ابتدای پروژه و مناسب موقعیت فعلی آن را پیکربندی نمود و بتوان در جریان اجرای فرایند پیکربندی را تغییر داد و اصلاح کرد.

گسترش پذیری:

*متدولوژی **RUP***: این متدولوژی به دلیل بزرگ بودن گسترش پذیر نیست و نقاط گسترش برای آن تعریف نشده و برعکس برای استفاده از آن باید آن را هرس کرد یا قطعات مورد نیاز را مجتمع کرد.

*متدولوژی **Fusion***: در این متدولوژی گسترش پذیری و مکانیزم های گسترش به صراحت تعریف نشده است اما برای استفاده کارا از آن باید آن را گسترش داد.

*متدولوژی **USDP***: این متدولوژی تعریف سطح بالایی دارد و قابلیت گسترش پذیری دارد و متدولوژی های زیادی با گسترش آن ایجاد شده اند.

*متدولوژی **EUP***: این متدولوژی به دلیل بزرگ بودن گسترش پذیر نیست و نقاط گسترش برای آن تعریف نشده و برعکس برای استفاده از آن باید آن را هرس کرد یا قطعات مورد نیاز را مجتمع کرد.

*مقایسه **RUP** با **Fusion***: در این زمینه با توجه با اینکه Fusion کمبود های بیشتری دارد می توان به راحتی آن را گسترش داد اما RUP بسیار پیچیده و بزرگ است و امکان گسترش آن وجود ندارد.

*مقایسه **RUP** با **USDP***: متدولوژی UP بهتر از RUP است چون سطح بالاتر تعریف شده و می تواند به خوبی گسترش پیدا کند.

*مقایسه **RUP** با **EUP***: در این زمینه دو متدولوژی مشابه هستند و گسترش آنها بسیار مشکل است. البته اینکه خود EUP گسترش یافته RUP است را شاید بتوان برتری RUP دانست.

مقیاس پذیری:

*متدولوژی **RUP***: این متدولوژی به دلیل داشتن فعالیت های مدیریتی و مدل های غنی می تواند تا حد خوبی مقیاس پذیر باشد همچنین پشتیبانی از سطحی از فرمالیزم و پشتیبانی ابزاری بالا هم باعث می شود بتوان برای گستره بزرگ تری از پروژه ها آن را به کار برد.

*متدولوژی **Fusion***: در این متدولوژی در صورتی که تعداد کلاس ها بالا باشد پیاده سازی آنها برای پروژه های سایز بزرگ و سطوح بحرانیت نیازمند ابزارهای کنترلی برای مدیریت کد و پیچیدگی ها خواهد بود. به همین خاطر هم Fusion Evo با اضافه کردن همین موارد معرفی شد تا بتواند پروژه های بزرگ مقیاس که نیازمند اجرای تکراری برای مدیریت ریسک هم بودند را به خوبی پوشش بدهد.

متدولوژی USDP: این متدولوژی به دلیل سطح بالا بودن می تواند برای سایز های مختلف سفارشی سازی شود که باعث می شود مقیاس پذیر باشد همچنین پشتیبانی از سطحی از فرمالیزم هم باعث می شود بتوان برای گستره بزرگ تری از پروژه ها آن را به کار برد. مدل های غنی هم می تواند به مقیاس پذیری آن کمک کند.

متدولوژی EUP: این متدولوژی به دلیل داشتن فعالیت های مدیریتی بسیار زیاد و مدل های بسیار غنی در همه سطوح می تواند تا حد خوبی مقیاس پذیر باشد البته به دلیل عدم اجبار به استفاده از UML ممکن است نتواند فرمالیزم داشته باشد که می تواند نوع پروژه هایی که می توان برای آن استفاده کرد را کاهش بدهد.

مقایسه RUP با Fusion: در RUP چون مدل های غنی تری دارد؛ فعالیت های مدیریت پروژه دارد و پشتیبانی ابزاری بالاتر و پشتیبانی از فرمالیزم دارد برای سایز ها و سطوح بحرانیت بیشتری می تواند استفاده شود و از این لحاظ برتر ازFusion است.

مقایسه RUP با USDP: با توجه به اینکه UP سطح بالاتر است و فعالیت های مدیریتی را کمرنگ تر پوشش می دهد RUP بهتر مقیاس می شود.

مقایسه RUP با EUP: در این زمینه از طرفی در EUP فعالیت های مدیریتی قوی تر، مدل ها غنی تر و پشتیبانی ابزاری خوبی دارد و از این لحاظ بهتر از RUP است. اما عدم پشتیبانی قطعی از فرمالیزم به دلیل عدم اجبار به استفاده از UML ممکن است باعث شود پشتیبانی از سطوح بحرانیت کمتری داشته باشد.

پیکربندی:

متدولوژی RUP: این متدولوژی قابلیت پیکربندی دارد و ابزار RMC را برای آن ایجاد کرده است اما به دلیل پیچیده بودن پیکربندی آن بسیار مشکل و هزینه بر است.

متدولوژی Fusion: در این متدولوژی صحبتی در مورد پیکربندی فرایند وجود ندارد.

متدولوژی USDP: این متدولوژی را می توان پیکربندی کرد اما ابزار خاصی برای پیکربندی آن معرفی نشده است.

متدولوژی EUP: این متدولوژی هم قابلیت پیکربندی دارد ولی به دلیل پیچیدگی بالا بسیار زمان بر، مشکل و نیازمند صرف هزینه زیاد است.

مقایسه RUP با Fusion: قابلیت پیکربندی با توجه به داشتن ابزار و اینکه Fusion پوششی برای آن ندارد در RUP بهتر است.

مقایسه RUP با USDP: در RUP چون ابزار وجود دارد و فعالیت ها گسترده تر هستند معیار های بیشتری هم برای پیکربندی وجود دارد و بهتر از UP است.

مقایسه RUP با EUP: در این زمینه دو متدولوژی مشابه هستند.

انعطاف پذیری:

متدولوژی RUP: در این متدولوژی در نظام محیط که در طول فازها اجرا می شود می توان فرایند را بهبود داد و یا تغییر داد. گاهی این کار در مرحله ارزیابی هم اتفاق می افتد.

متدولوژی Fusion: در این متدولوژی فعالیتی برای بازبینی یا تغییر فرایند در میانه اجرا تعریف نشده است.

متدولوژی USDP: در این متدولوژی نظام خاص و فعالیت خاصی برای این کار نداریم . در آن تعبیه نشده است.

متدولوژی EUP: در این متدولوژی در قالب نظام بهبود فرایند نرم‌افزار می توان فرایند را بهبود بخشید که به شکل جلسه یا کارگاه برگزار می شود و به شکل فرایند دائمی در طول اجرا انجام می شود و فرایند ایجاد نرم‌افزار در سازمان را بهبود می بخشد.

مقایسه RUP با Fusion: در RUP نظام خاص برای این کار تعریف شده و از Fusion که فعالیتی برای این کار ندارد بهتر است.

مقایسه RUP با USDP: تغییرات در میانه فرایند در UP کمرنگ است ولی RUP نظام مشخص برای این کار دارد و در این زمینه بهتر از UP است.

مقایسه RUP با EUP: در این زمینه دو متدولوژی مشابه هستند اما فعالیت هایی که EUP انجام می دهد گسترده تر هستند و از این لحاظ می توان EUP را بهتر دانست.

معیار حوزه کاربرد:

متدولوژی در چه حوزه هایی کاربرد دارد و برای چه حوزه هایی مناسب نیست حداقل باید بتواند حوزه کاربرد سیستم های اطلاعاتی را پوشش دهد.

متدولوژی RUP: این متدولوژی را می توان برای انواع مختلف پروژه ها استفاده کرد و با توجه به داشتن سطحی از فرمالیزم ، مدل های غنی و ابزارهای متنوع حوزه کاربردش می تواند گسترش پیدا کند .

متدولوژی Fusion: این متدولوژی برای انواع مختلف سیستم ها هم در خود HP و هم در دیگر سازمان ها با موفقیت تست شده است. از جمله سیستم های می توان به سیستم های مربوط به شبکه، سیستم های بلادرنگ، سیستم های اطلاعاتی و مدیریت ویدیو اشاره نمود اما خودش پیشنهاد می کند از آن برای سیستم هایی که همزمانی دارند و سیستم های توزیع شده به دلیل عدم مدل سازی اینها در فرایند استفاده نشود. همچنین به دلیل عدم وجود فرمالیزم با حد بالا برای سیستم های با بحرانیت Life مناسب نمی باشد.

متدولوژی USDP: این متدولوژی را می توان برای انواع مختلف پروژه ها استفاده کرد و با توجه به داشتن سطحی از فرمالیزم ، مدل های غنی و ابزارهای متنوع حوزه کاربردش می تواند گسترش پیدا کند .

متدولوژی EUP: این متدولوژی را می توان برای انواع مختلف پروژه ها استفاده کرد و با توجه به نداشتن فرمالیزم به خاطر عدم اجبار به استفاده از UML حوزه کاربردش می تواند کمتر باشد و در این متدولوژی برای پروژه های با کاربردهای مختلف در سطح سازمان ها استفاده می شود. البته مدل های متنوع و غنی و ابزارهای پشتیبان می تواند دامنه کاربردش را گسترش بدهد.

مقایسه RUP با Fusion: فرمالیزم، مدل های غنی تر و ابزارهای بیشتر باعث می شود حوزه کاربرد RUP بیشتر از Fusion باشد.

مقایسه RUP با USDP: در این زمینه دو متدولوژی مشابه هستند ولی اینکه RUP پشتیبانی ابزاری بهتری دارد می تواند برای حوزه های بیشتری استفاده شود.

مقایسه RUP با EUP: نداشتن فرمالیزم به طور قطعی می تواند حوزه های کاربرد EUP را کاهش دهد ولی از طرفی داشتن فرایندهای سازمانی کمک می کند دامنه کاربرد ان گسترش پیدا کند. در صورت استفاده از UML که می تواند فرمالیزم را اضافه کند در مجموع EUP بهتر از RUP است.

معیارهای زبان مدل سازی:

معیار پشتیبانی از مدل سازی شی گرا سازگار، دقیق و بدون ابهام:

مدل ها باید با کیفیت و شامل هر چه در یک مدل شی گرا نیاز است باشد و بتوان به وضوح شی گرایی را در آن دید، مدل ها نباید یکدیگر را نقض کنند، مدل باید صحیح باشند و اشتباه نداشته باشند همچنین باید با جزئیات کامل باشد و زبان مدل سازی اینها را پشتیبانی کند.

دیدگاه های مختلف مدل سازی:

متدولوژی RUP: در این متدولوژی هر سه دید وظیفه ای، رفتاری و ساختاری پوشش داده می شود و به دلیل استفاده از UML مدل ها می توانند غنی باشند. مدل سازی از قلمرو مسئله آغاز می شود و تا کلاس ها ادامه پیدا می کند. Business Usecase Model و Usecase Model وظیفه ای، Business Process Model، Activity Diagram، Sequence Diagram و Timing Diagram رفتاری و Business Object Model، Class Diagram، Component Diagram، Package Diagram و Deployment Diagram مدل های ساختاری هستند.

متدولوژی Fusion: در این متدولوژی هر سه دید وظیفه ای، رفتاری و ساختاری پوشش داده می شود. البته رفتار داخل کلاس ها مدل نمی شوند و در فاز تحلیل مدل سازی وظیفه ای و رفتاری در مرز سیستم است و داخل سیستم نیست. Transaction Scenario و Object Interaction Graph، Visibility Graph و Inhertance Graph رفتاری، Object Model ساختاری و System Interface وظیفه ای است.

متدولوژی USDP: در این متدولوژی هر سه دید وظیفه ای، رفتاری و ساختاری پوشش داده می شود و به دلیل استفاده از UML مدل ها می توانند غنی باشند. مدل سازی از قلمرو مسئله آغاز می شود و تا کلاس ها ادامه پیدا می کند. Usecase Model وظیفه ای، Activity Diagram، Sequence Diagram و Timing Diagram رفتاری و Class Diagram، Component Diagram، Package Diagram و Deployment Diagram مدل های ساختاری هستند.

متدولوژی EUP: در این متدولوژی هر سه دید وظیفه ای، رفتاری و ساختاری پوشش داده می شود به دلیل عدم اجبار به استفاده از UML می توان مدل ها را با مدل هایی که برای سازمان مناسب تر است هم غنی نمود. مدل سازی از سطح کسب‌وکار تا کلاس ها در این متدولوژی انجام می شود. Business Usecase Model و Usecase Model وظیفه ای، Business Process Model، Activity Diagram، Sequence Diagram و Timing Diagram رفتاری و Business Object Model، Class Diagram، Component Diagram، Package Diagram و Deployment Diagram مدل های ساختاری هستند.

مقایسه RUP با Fusion: در این زمینه RUP با پوشش کامل مدل سازی در همه سطوح بهتر از Fusion می باشد.

مقایسه RUP با USDP: در این زمینه دو متدولوژی مشابه هستند.

مقایسه RUP با EUP: در این زمینه دو متدولوژی مشابه هستند.

مدل سازی از منطقی به فیزیکی:

متدولوژی RUP: در این متدولوژی مدل سازی از use case شروع می شود اما تحلیل در آن اجباری نیست ولی طراحی را به خوبی انجام می دهد.

متدولوژی Fusion: در این متدولوژی با اضافه کردن کلاس های مربوط به پیاده سازی به کلاس های تحلیل مدل سازی فیزیکی انجام می شود.

متدولوژی USDP: این متدولوژی مدل سازی از منطقی به فیزیکی را تکمیل کردن مدل های تحلیل در طراحی انجام می دهد و کامل تحلیل و طراحی را پوشش می دهد.

متدولوژی EUP: این متدولوژی مدل سازی از منطقی به فیزیکی را تکمیل کردن مدل های تحلیل در طراحی انجام می دهد و کامل تحلیل و طراحی را پوشش می دهد.

مقایسه RUP با Fusion: طراحی در RUP اجباری نیست که برای آن ضعف است ولی تا سطح پیاده سازی را پوشش می دهد که در کل بهتر از Fusion است.

مقایسه RUP با USDP: چون تحلیل اختیاری است در این زمینهUP بهتر از RUP می باشد.

مقایسه RUP با EUP: در این زمینه دو متدولوژی مشابه هستند.

سطوح مختلف انتزاع و دانه بندی:

متدولوژی RUP: مدل سازی را از سطح کسب‌وکار آغاز می کند و تا ریز ترین سطح که داخل کلاس ها هستند را مدل سازی می کند و سطوح مختلف را پوشش می دهد.

متدولوژی Fusion: در این متدولوژی مدل ها از قلمرو مسئله آغاز و تا سطح کلاس ها ادامه می یابد. البته این کار با شکستن مدل ساختاری در تحلیل انجام می شود. اما به طور کلی از سطوح بالایی از دانه بندی پشتیبانی نمی کند.

*متدولوژی **USDP***: مدل سازی را از سطح کسب‌وکار آغاز می کند و تا ریز ترین سطح که داخل کلاس ها هستند را مدل سازی می کند و سطوح مختلف را پوشش می دهد. . از پکیج ها می توان برای این کار استفاده نمود.

*متدولوژی **EUP***: مدل سازی را از سطح کسب‌وکار آغاز می کند و تا ریز ترین سطح که داخل کلاس ها هستند را مدل سازی می کند و سطوح مختلف را پوشش می دهد. . از پکیج ها می توان برای این کار استفاده نمود.

*مقایسه **RUP** با **Fusion***: در RUP سطوح مختلف را پوشش می دهد ولی Fusion در تحلیل سطح سیستم را پوشش نمی دهد و در مرز سیستم است که در مجموع RUP بهتر است.

*مقایسه **RUP** با **USDP***: در این زمینه دو متدولوژی مشابه هستند.

*مقایسه **RUP** با **EUP***: در این زمینه دو متدولوژی مشابه هستند.

مدل سازی صوری و غیر صوری:

*متدولوژی **RUP***: این متدولوژی برای مدل سازی از UML استفاده می کند که مدل بصری تولید می کند همچنین می توان با کمک OCL مدل سازی صوری هم انجام داد. همچنین مدل های متنی زیادی هم در طول فرایند انجام می شود.

*متدولوژی **Fusion***: این متدولوژی زبان مدل سازی بصری دارد اما مختص به خودش نیست. برای مدل سازی های متنی هم قواعدی دارد و همچنین با کمک عبارات منظم تا حدی مدل سازی صوری هم انجام می دهد.

*متدولوژی **USDP***: این متدولوژی برای مدل سازی از UML استفاده می کند که مدل بصری تولید می کند همچنین می توان با کمک OCL مدل سازی صوری هم انجام داد. همچنین مدل های متنی زیادی هم در طول فرایند انجام می شود.

*متدولوژی **EUP***: مدل سازی غیر صوری می تواند با هر مدل انجام شود. به علاوه مدل سازی صوری هم می تواند انجام شود ولی خود متدولوژی توصیه ای درمورد مدل سازی صوری نداشته است.

*مقایسه **RUP** با **Fusion***: به دلیل داشتن مدل های بهتر و پشتیبانی از مدل های صوری RUP بهتر از Fusion است.

*مقایسه **RUP** با **USDP***: در این زمینه دو متدولوژی مشابه هستند.

مقایسه RUP با EUP: از لحاظ اینکه RUP مدل سازی صوری را به طور قطعی پوشش می دهد در مجموع بهتر از EUP است.

معیار مدیریت تناقض ها و پیچیدگی ها:

پیچیدگی مدل ها باید مدیریت شود و از بوجود آمدن ناسازگاری بین آنها جلوگیری نمود. زبان مدل سازی باید برای تشخیص و رفع ناسازگاری و پیچیدگی مدل ها استراتژی و تکنیک های مناسب داشته باشد.

متدولوژی RUP: برای رفع ناسازگاری ها در این متدولوژی خود UML با کمک مکانیزم هایش می تواند کمک کننده باشد. همچنین برای مدیریت پیچیدگی مدل ها می توان از پکیج ها برای بخش بندی و لایه بندی استفاده نمود.

متدولوژی Fusion: در این متدولوژی برای اینکه همه چیز سازگار پیش برود چک لیست هایی در انتهای هر فاز دارد همچنین Data Dictionary هم به سازگاری مدل ها کمک می کند. اما خود متدولوژی برای مدیریت پیچیدگی ها فعالیت خاصی ندارد. با کمک ابزار Paradigm Plus می توان بررسی سازگاری را با کمک نوشتن اسکریپت انجام داد. می توان کلاس ها را به کمک جدا سازی نمودار ها و به کمک قواعد موجود در متدولوژی که کمک به اشتراک گذاری کلاس ها و رابطه ها می دهد و آن را از OMT گرفته است خوشه بندی کرد که می تواند به مدیریت پیچیدگی تا حدی کمک نماید.

متدولوژی USDP: برای رفع ناسازگاری ها در این متدولوژی خود UML با کمک مکانیزم هایش می تواند کمک کننده باشد. همچنین برای مدیریت پیچیدگی مدل ها می توان از پکیج ها برای بخش بندی و لایه بندی استفاده نمود.

متدولوژی EUP: همچنین برای مدیریت پیچیدگی مدل ها می توان از پکیج ها برای بخش بندی و لایه بندی استفاده کند . به دلیل عدم اجبار در استفاده از UML مدیریت پیچیدگی ها و ناسازگاری ها در آن بسیار مشکل است.

مقایسه RUP با Fusion: به دلیل استفاده از UML و پشتیبانی ابزاری قوی RUP بهتر می تواند پیچیدگی ها و ناسازگاری را مدل کند.

مقایسه RUP با USDP: از جهت داشتن پشتیبانی ابزاری قوی تر RUP بهتر از UP در این زمینه است.

مقایسه RUP با EUP: در این زمینه دو متدولوژی مشابه هستند اما اجبار به استفاده از UML در RUP باعث می شود پیچیدگی ها و ناسازگاری بهتر مدیریت شوند.