



فصل دوم

آشنایی با فرآیندهای نرم افزار

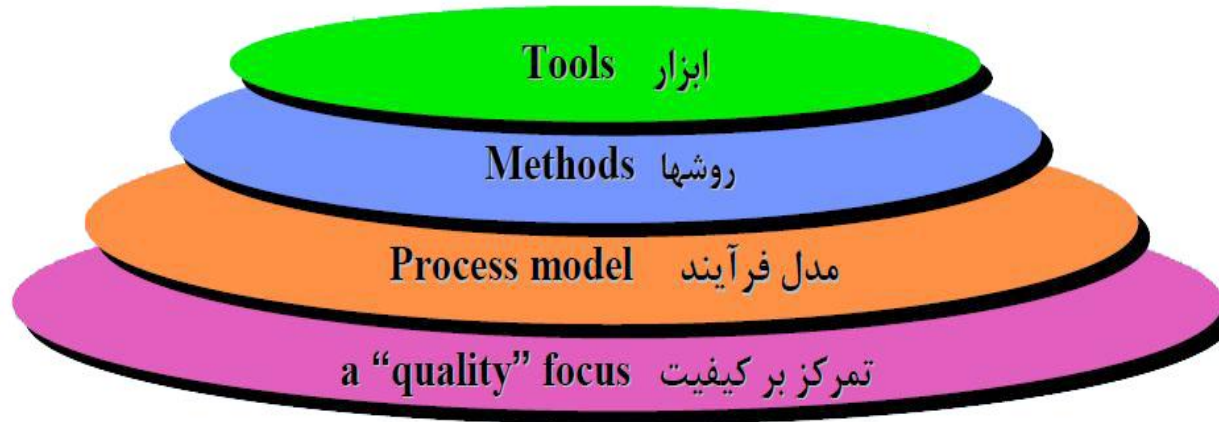
تعریف فرآیند نرم افزاری

فرآیند نرم افزاری به عنوان چارچوب کاری تعریف می شود که برای ایجاد یک نرم افزار دارای کیفیت مطلوب لازم است.



تولید نرم افزار کامپیوتری، یک فرآیند یادگیری تکرار شونده است. حاصل آن چه که سرمایه نرم افزاری نام نهاده اند، قرار دادن دانش جمع آوری شده، گزینش و سازماندهی آن به عنوان فرآیند در دست اقدام است.

مهندسی نرم افزار: یک فناوری لایه ای



- ✓ هرگونه رهیافت مهندسی باید بر یک تعهد سازمانی در مورد کیفیت تکیه داشته باشد.
- ✓ فرآیند چارچوبی برای مجموعه ای از حوزه های کلیدی و اصلی مهیا می سازد که باید برای تحویل موثر فناوری مهندسی نرم افزار به وجود آید.
- ✓ روش های مهندسی نرم افزار به مجموعه ای از اصول تکیه دارند که بر هر حوزه فناوری حاکم می باشد.
- ✓ ابزارهای مهندسی نرم افزار پشتیبانی تمام خودکار یا نیمه خودکار را برای فرآیند و روش ها فراهم می سازند.

سطوح بلوغ فرآیند

سطح ۱: آغاز

✓ فرآیند نرم افزاری به صورت موقتی و حتی گاه به صورت بسیار آشفته توصیف می شود. در این مرحله چند فرآیند تعریف می شود. موفقیت به تلاش های فردی بستگی دارد.

✓ کوشش چندانی روی تضمین کیفیت و مدیریت پروژه صورت نمی پذیرد.

✓ هر تیم پروژه مجاز به ایجاد نرم افزار به هر شیوه و با استفاده از هرگونه روش استاندارد و رویه ای است که ممکن است از خوب تا بسیار ضعیف عمل کند.

سطوح بلوغ فرآیند

سطح ۲: قابل تکرار

✓ فرآیندهای اولیه مدیریت پروژه برای مشخص کردن هزینه، زمان بندی و قابلیت کارایی آن صورت می گیرد. اصل ضروری این سطح در جایی است که موفقیت های پیش تر پروژه هایی با کاربرد مشابه را تکرار می کند.

✓ نمایان گر این حقیقت است که ارائه دهنده نرم افزار، فعالیت های خاصی چون گزارش تکمیل کار، گزارش زمان و کارهای صورت گرفته را تعریف کرده است.

سطوح بلوغ فرآیند

سطح ۳: تعریف شده

✓ فرآیند نرم افزاری برای مدیریت و فعالیت های مهندسی در طول یک فرآیند پردازش نرم افزاری در سازمان ثبت شده، استاندارد و منسجم است.

✓ همه پروژه ها از یک نسخه تأیید شده و ثبت شده در فرآیند پردازش، برای توسعه و پشتیبانی نرم افزار استفاده می کنند.

✓ این سطح بیانگر این حقیقت است که ارائه دهنده نرم افزار، هم فرآیندهای مدیریتی و هم فنی را تعیین کرده است.

✓ در این سطح است که ارائه دهندگان نرم افزار اقدام به اخذ استانداردهایی از قبیل ISO 9001 می کنند.

سطوح بلوغ فرآیند

سطح ۴: مدیریت شده

✓ موازین دقیقی از فرآیند و کنترل محصول جمع آوری می شود. هم فرآیند نرم افزار و هم محصولات از لحاظ کمی درک می شوند و با استفاده از این موازین دقیق کنترل می گردند.

✓ این سطح در بر گیرنده مفهوم اندازه گیری و استفاده از استانداردها و معیارهاست. معیار یک کمیت معنادار است که با استفاده از آن می توان از یک سند یا کد موجود در پروژه نرم افزاری، اطلاعاتی را استخراج نمود.

✓ سازمانی که در این سطح قرار دارد معیارهای متعددی را جمع آوری کرده، سپس از آن ها برای نظارت بر پروژه و کنترل آن استفاده می نماید.

سطوح بلوغ فرآیند

سطح ۵: بهینه شده

✓ بهبود مکرر فرآیند با توجه به بازخورد کمی و از روی آزمون ایده های نوآورانه و فناوری های تازه، مقدر است.

✓ این سطح، بالاترین سطح قابل دستیابی است که تاکنون تنها تعداد ناچیزی از سازندگان نرم افزار به این مرحله رسیده اند. سازنده نرم افزار در این مرحله قادر به پیش بینی پیامدهایی از قبیل تعداد اشکال های باقیمانده در محصول، بر اساس سنجش های انجام شده در حین انجام پروژه است.

مدل های فرآیند

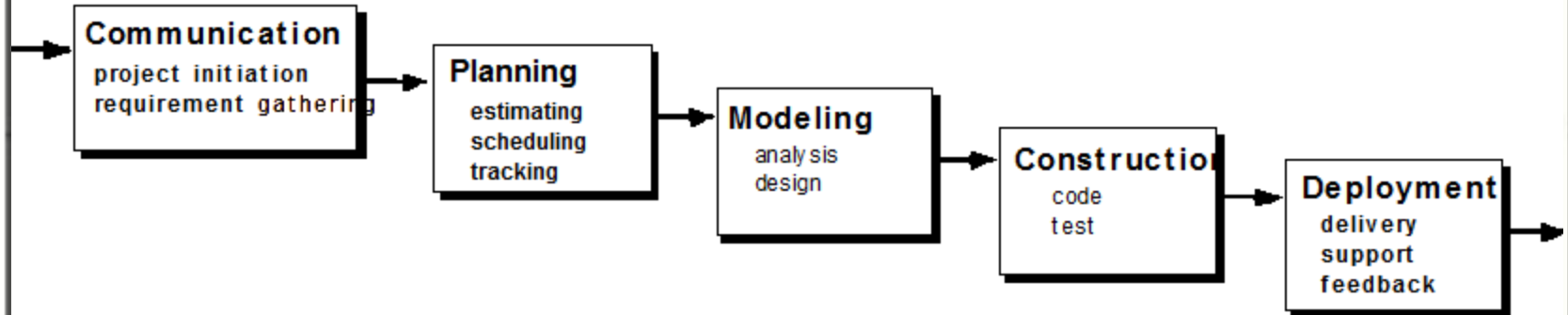
انتخاب مدل فرآیند بر اساس نوع نرم افزاری که قصد توسعه آن را داریم و نیز طبیعت پروژه و کاربرد آن صورت می پذیرد.

مدل های مختلف در فرآیند تولید نرم افزار عبارتند از:

۱. مدل آبشاری
۲. مدل نمونه سازی
۳. مدل توسعه سریع نرم افزار
۴. مدل افزایشی
۵. مدل حلزونی

مدل آبشاری waterfall model

مدل آبشاری مدلی سنتی و متداول است که به آن مدل چرخه حیات کلاسیک یا مدل ترتیبی خطی نیز گفته می شود.



در این مدل، توسعه نرم افزار مجموعه روال ترتیبی از فرآیندهای موجود در

مهندسی نرم افزار را طی می کند تا به نتیجه نهایی برسد.

مزایای مدل آبشاری

۱. به دلیل تفکیک فعالیت ها به سیستم های مختلف کاری، مدیریت پروژه ساده است.
۲. ارائه اصول و تکنیک ها در اکثر مراحل این مدل، انجام مراحل را آسان می کند؛ علاوه بر آن باعث افزایش کیفیت و دید نرم افزار می گردد.
۳. مدل، مستندگرا است و مستندات خروجی هر مرحله، ورودی مرحله بعد می باشد.
۴. مراحل این مدل، انطباق زیادی با مراحل کلی که در همه نمونه های مهندسی نرم افزار به کار می رود، دارند.

معایب مدل آبشاری

۱. پروژه های واقعی به ندرت از جریان زنجیری و متوالی مدل پیروی می کنند. در نتیجه با پیش رفتن تیم پروژه، ممکن است تغییرات باعث سردرگمی شوند.

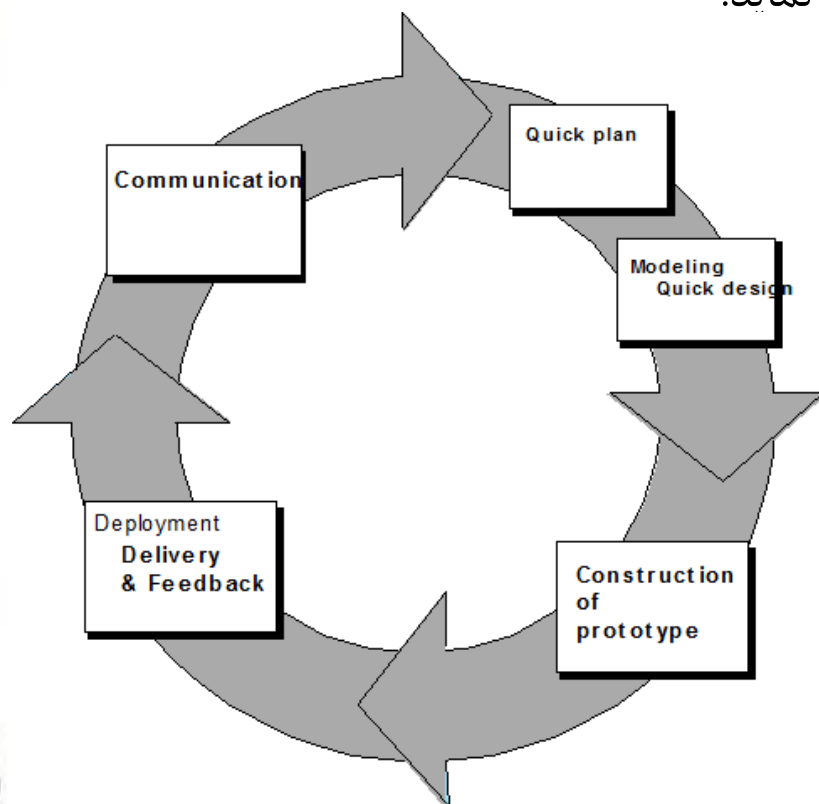
۲. اغلب برای مشتری مشکل است که تمام نیازهای خود را به طور مشخص بیان کند. مدل آبشاری به بیانی واضح نیاز داشته و به خوبی پاسخ گوی موارد غیر قطعی که در آغاز پروژه ها وجود دارد، نمی باشد.

۳. مشتری باید صبر زیادی داشته باشد. اولین نسخه کاری از پروژه معمولاً تا اواخر مدت کاری در دسترس او قرار نخواهد گرفت.

۴. اگر اشتباهی وجود داشته باشد، این اشتباه تا اواخر کار برنامه نویسی که پروژه مورد بازبینی قرار می گیرد، مشخص نمی گردد و این می تواند تهدیدی فاجعه آمیز باشد.

مدل نمونه سازی prototyping model

نمونه سازی فرآیندی است که توسعه دهنده را قادر به ایجاد یک مدل از نرم افزار مورد نظر می نماید.



نحوه به کارگیری مدل:

- الگوی ساخت نمونه اولیه با جمع آوری نیازمندیها آغاز می شود.
- یک طراحی سریع صورت می گیرد.
- طراحی سریع منجر به ساخت نمونه اولیه می شود
- نمونه اولیه مورد ارزیابی کاربر قرار می گیرد.
- با تنظیم نمونه اولیه برای برآوردن نیازهای مشتری تکرار رخ می دهد.

مدل نمونه سازی prototyping model

مزایای مدل نمونه سازی

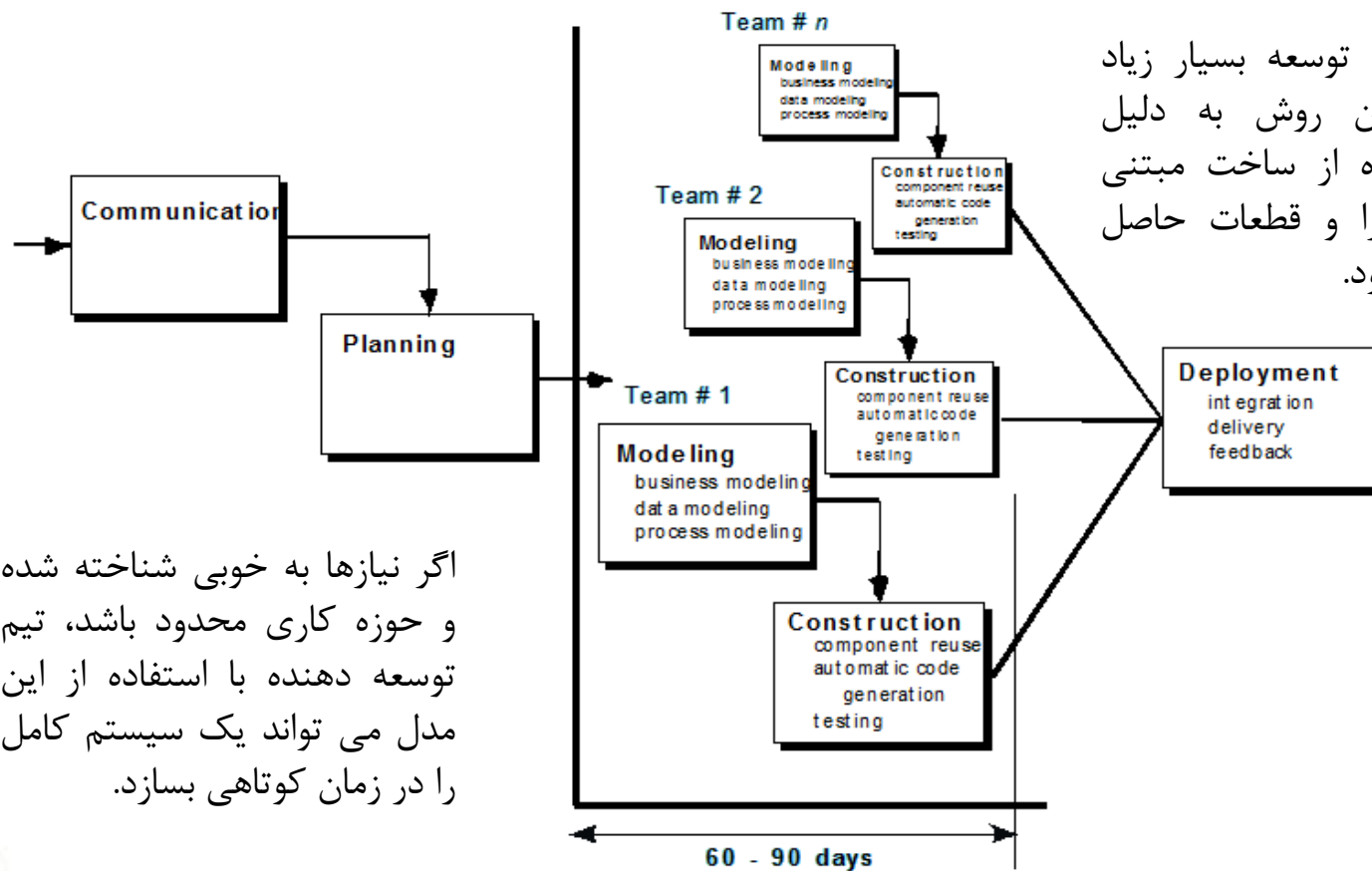
۱. در تمام طول کار، ارتباط با مشتری برقرار است.
۲. تغییر نیازها امکان پذیر می باشد.

معایب مدل نمونه سازی

۱. از آن جا که کارفرما نسخه اولیه را دیده و پسندیده است، ممکن است زمان زیادی را که برای تکمیل کار نیاز است به راحتی نپذیرد.
۲. سرعت انجام کار باعث اثرات جانبی می گردد.

مدل توسعه سریع نرم افزار RAD model

یک مدل فرآیند تولید نرم افزار است که تنها بر یک سیکل تولید بسیار کوتاه تأکید می کند.



سرعت توسعه بسیار زیاد در این روش به دلیل استفاده از ساخت مبتنی بر اجزا و قطعات حاصل می شود.

اگر نیازها به خوبی شناخته شده و حوزه کاری محدود باشد، تیم توسعه دهنده با استفاده از این مدل می تواند یک سیستم کامل را در زمان کوتاهی بسازد.

مدل توسعه سریع نرم افزار RAD model

این مدل شامل مراحل زیر است:

■ مدل سازی تجاری

■ مدل سازی داده ای

جریان اطلاعات در میان کارهای تجاری به شکلی مدل سازی می شود که به سوالات زیر پاسخ دهد:

چه اطلاعاتی برای پیشبرد فرآیند تجاری نیاز است؟

چه اطلاعاتی تولید می شوند؟

چه کسانی این اطلاعات را تولید می کنند؟

این اطلاعات به کجا می روند؟

چه کسانی آن را پردازش می کنند؟

مدل توسعه سریع نرم افزار RAD model

این مدل شامل مراحل زیر است:

▪ مدل سازی تجاری

▪ مدل سازی داده ای

▪ مدل سازی فرآیند

جریان اطلاعاتی به دست آمده در مرحله قبل به صورتی کامل پالایش می شود و اشیاء داده ای مورد نیاز از آن استخراج می گردند. صفات هر شی مشخص شده و ارتباط بین اشیا مشخص می گردد.

مدل توسعه سریع نرم افزار RAD model

این مدل شامل مراحل زیر است:

▪ مدل سازی تجاری

▪ مدل سازی داده ای

▪ مدل سازی فرآیند

▪ تولید برنامه کاربردی

پردازش ها و فرآیندهای مورد نیاز برای ساخت، اضافه کردن، تغییر، حذف و یا بازیابی اشیاء داده ای تعریف می شوند.

مدل توسعه سریع نرم افزار RAD model

این مدل شامل مراحل زیر است:

▪ مدل سازی تجاری

▪ مدل سازی داده ای

▪ مدل سازی فرآیند

▪ تولید برنامه کاربردی

▪ آزمون و چرخش

این مدل فرض را به استفاده از تکنیک های تولید نسل چهارم گذاشته است.

در این حال به جای نوشتن کد از اشیا و قطعات از قبل آماده شده استفاده می شود و در صورت امکان اجزای قابل استفاده مجددی تولید می گردد. در همه موارد، ابزارهای خودکار برای تسهیل در ساخت نرم افزار به کار گرفته می شوند.

مدل توسعه سریع نرم افزار RAD model

این مدل شامل مراحل زیر است:

▪ مدل سازی تجاری

▪ مدل سازی داده ای

▪ مدل سازی فرآیند

▪ تولید برنامه کاربردی

▪ آزمون و چرخش

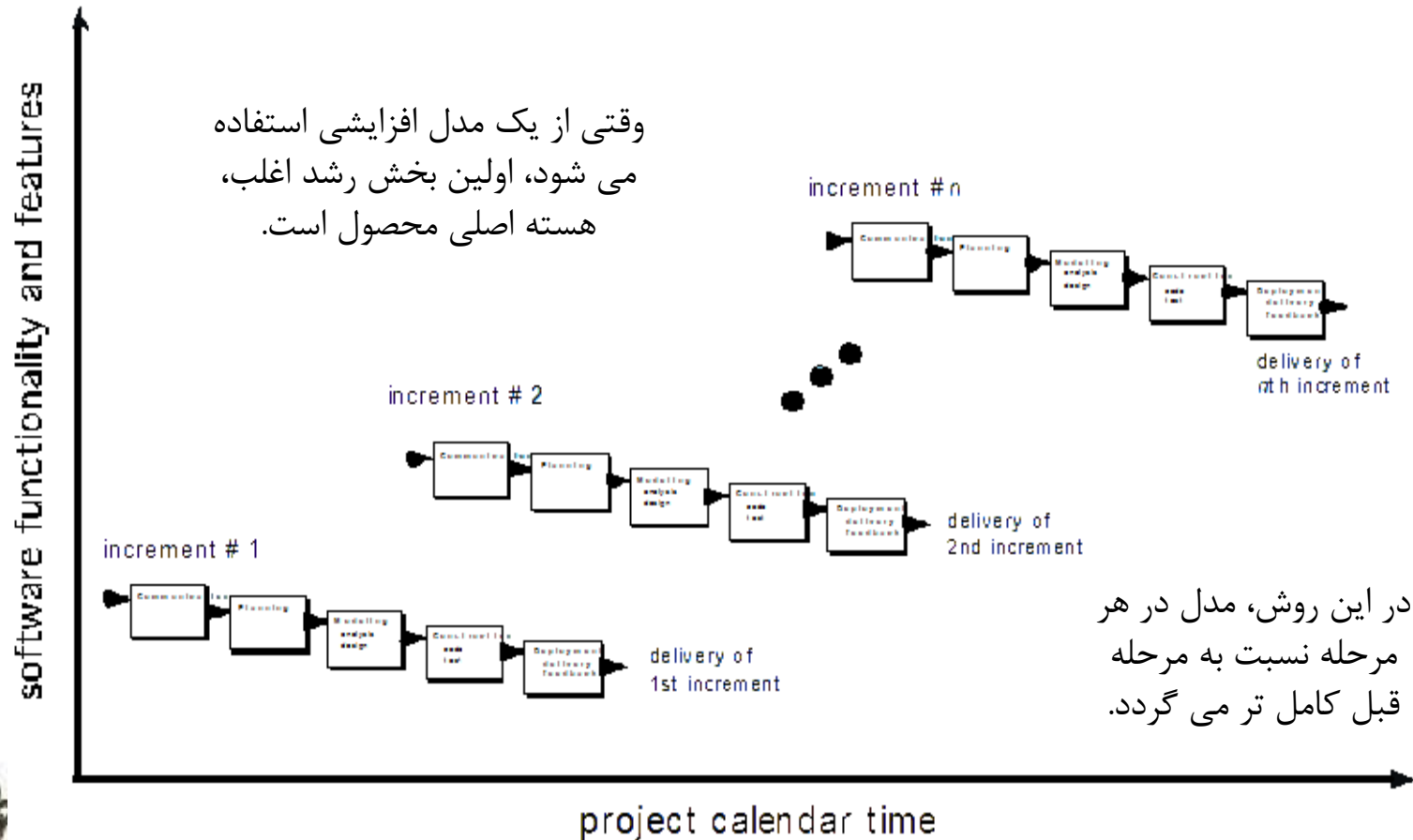
از آن جا که این مدل بر استفاده مجدد از اجزا تأکید دارد و بسیاری از مؤلفه ها قبلاً مورد آزمایش قرار گرفته اند، زمان تست کاهش می یابد؛ اما در این قسمت باید اجزا جدید و رابط های اجزا مورد تست قرار گیرند.

معایب مدل توسعه سریع نرم افزار

۱. در پروژه های بزرگ ولی مقیاس پذیر، برای ایجاد تعداد مناسبی از تیم های RAD، باید نیروی انسانی کافی در اختیار داشته باشیم.
۲. RAD نیاز به توسعه دهندگان و مشتریانی دارد که نسبت به فعالیت های سریع مورد نیاز برای کامل کردن سیستم در یک چارچوب زمانی فشرده تر متعهد باشند. اگر هر یک از طرفین فاقد تعهد لازم باشند، پروژه به شکست خواهد انجامید.
۳. این مدل، در همه انواع کاربردها قابل اعمال نیست.
۴. RAD در جایی که برنامه کاربردی از یک فناوری نوین استفاده نموده و ریسک فنی بالا می باشد چندان مناسب نیست.

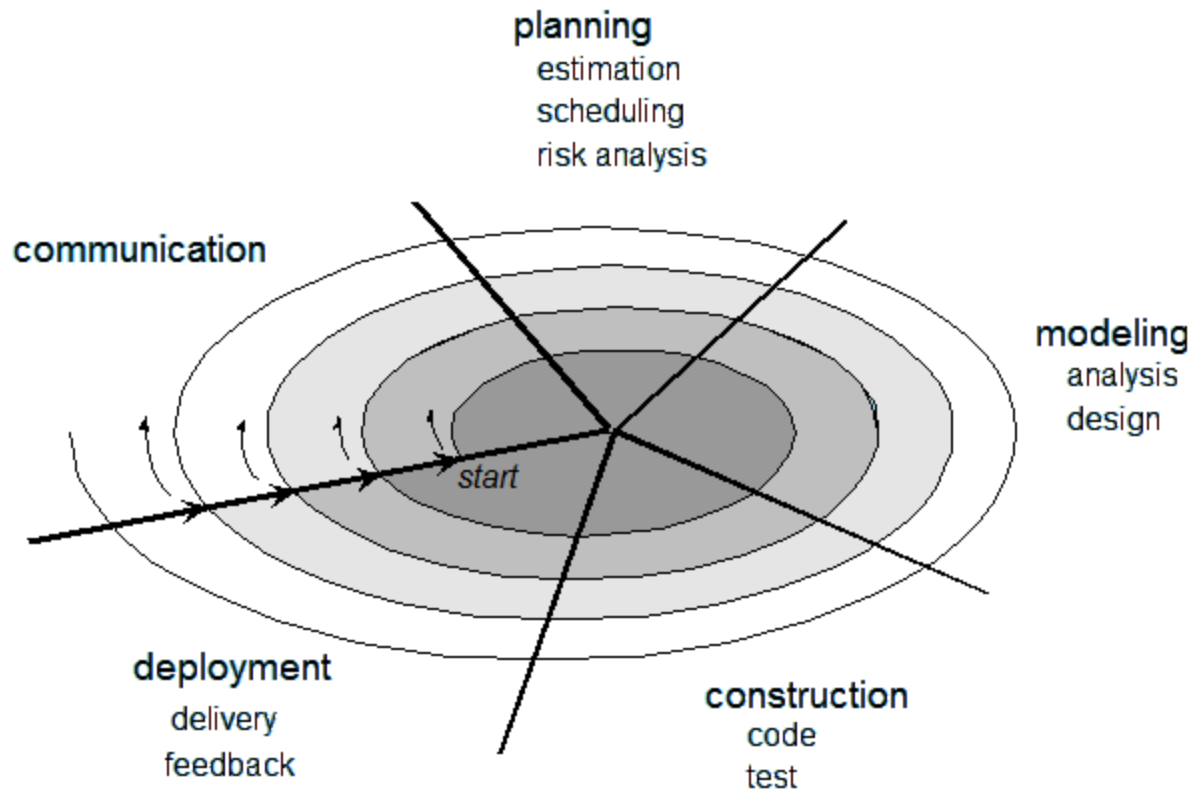
مدل افزایشی incremental model

در مدل افزایشی، مراحل مدل آبخاری با فلسفه گردشی مدل نمونه سازی ترکیب شده اند.



مدل حلزونی spiral model

مدل حلزونی یک مدل تکاملی است که ماهیت تکراری ساخت نمونه اولیه را با جنبه های سیستماتیک و کنترلی مدل آشنایی ترکیب می کند.



معایب مدل حلزونی

۱. ممکن است به سختی بتوان مشتری را قانع کرد (به ویژه در شرایط قرارداد) که روش تکاملی قابل کنترل است.

۲. این مدل مهارت ارزیابی خطر فراوانی را طلب می کند و برای موفقیت بر همین مهارت متکی است.

۳. مدل فوق به اندازه مدل های آبشاری و ساخت نمونه اولیه به کار گرفته نشده است و سال ها زمان لازم است تا بازدهی این الگوی جدید و مهم با قطعیت مطلق تعیین گردد.