

مهندسی نرم افزار
تجزیه و تحلیل و مدلسازی سیستمهای کامپیوتری

مراجع:

تحلیل و طراحی سیستم ها دکتر سعید پارسا
S/W Engineering Pressman 2001

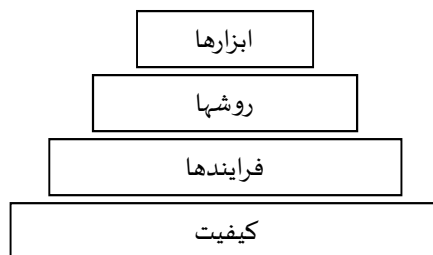
فصل

۲

- تعریف فرآیند
- فعالیتهای مهندسی نرم افزار
- مدل‌های فرایند
- فرآیند RUP
- موارد سختی تولید نرم افزار

فرایند

- فرایند نرم افزار نقشه راهی (Road Map) است که دو هدف زیر را دنبال می نماید:
 - کیفیت بالا
 - زمانبندی مناسب
- لایه های مهندسی نرم افزار



اساس مهندسی نرم افزار لایه فرایند می باشد.

فعالیت‌های مهندسی نرم افزار

• بطور کلی فعالیت‌های مربوط به مهندسی نرم افزار در سه فاز زیر دسته بندی می گردد:

▪ فاز تعریف

▪ فاز توسعه

▪ فاز پشتیبانی

نگهداشت تصحیحی

نگهداشت تطبیقی

نگهداشت بهبودی

نگهداشت پیشگیرانه

• فازهای فوق با یکسری فعالیت‌های چتری (Umbrella activities) تکمیل می گردد. مهمترین آنها عبارتند از:

کنترل و ردیابی نمودن پروژه های نرم افزاری ، تضمین کیفیت نرم افزار ، مدیریت پیکربندی نرم افزار ، تهیه مستندات ، مدیریت قابلیت استفاده مجدد ، سنجش و مدیریت ریسک

مدلهای فرایند

• تعیین یک راهکار توسعه که شامل لایه های فرایند ، روشها ، ابزار و فازها (تعریف ، توسعه و پشتیبانی) باشد را الگوی مهندسی نرم افزار یا مدل فرایند می نامند.

• مدل‌های فرایند نرم افزار عبارتند از :

▪ مدل ترتیبی خطی

▪ مدل ایجاد نمونه اولیه (Prototyping Model)

▪ مدل RAD

▪ مدل افزایشی (Incremental Model)

▪ مدل حلزونی (Spiral Model)

▪ مدل حلزونی برنده - برنده (Win-Win)

▪ مدل توسعه همزمان

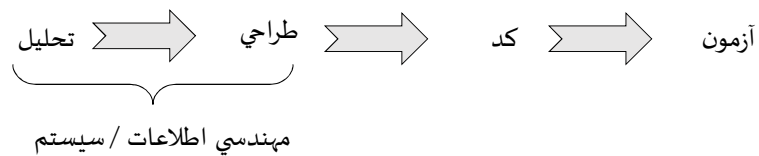
▪ مدل توسعه مبتنی بر مولفه (Component Base Development)

▪ مدل روشهای رسمی (Formal Method)

▪ تکنیکهای نسل چهارم

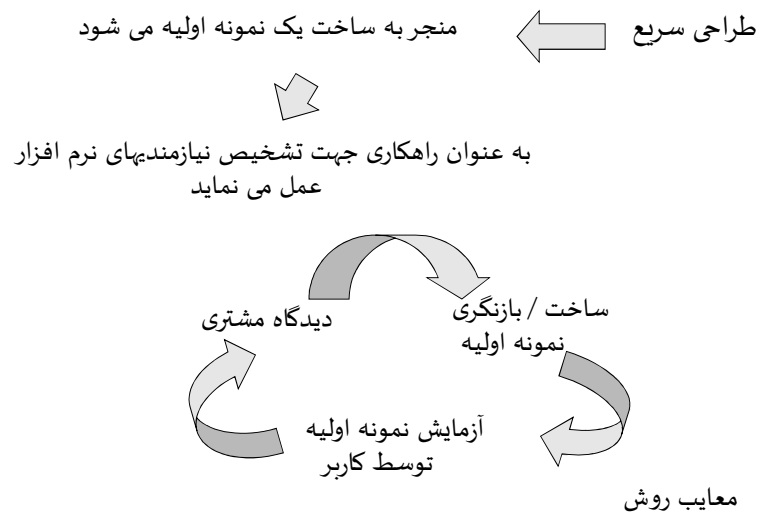
▪ مدل RUP

مدل ترتیبی خطی (مدل آبشاری ، چرخه حیات کلاسیک)



- طراحی نرم افزار بر روی چهار مورد زیر تمرکز می نماید :
 - ساختمان داده ها
 - معماری نرم افزار
 - نمایش واسط ها
 - الگوریتم ها
- معایب روش

مدل ساخت نمونه اولیه

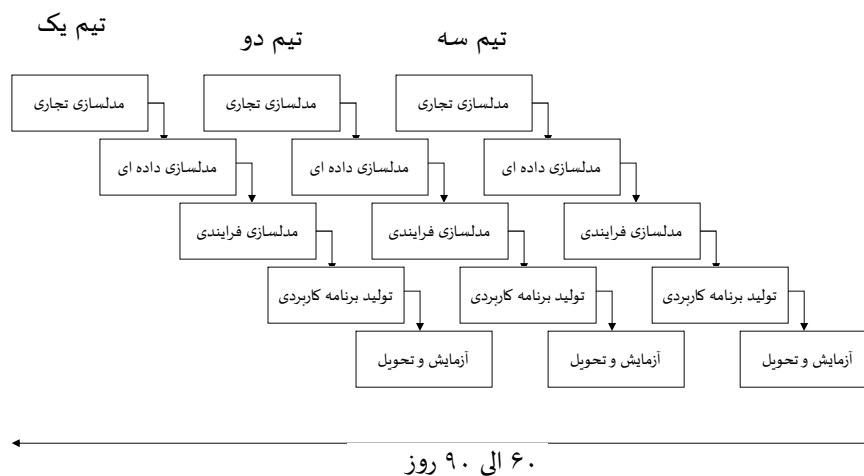


مدل RAD

- مدل توسعه کاربردی سریع (Rapid Application Development) یک فرایند توسعه تدریجی نرم افزار می باشد که بر ساخت مبتنی بر مولفه تاکید می نماید.
- فازهای مدل مذکور عبارتند از:
 - مدلسازی کاری
 - مدلسازی داده ای
 - مدلسازی فرایند
 - تولید برنامه کاربردی
 - آزمایش
- چنانچه بتوان سیستم را به چند زیر سیستم تقسیم نمود به گونه ای که توسعه هر زیر سیستم حداکثر ظرف مدت سه ماه به اتمام برسد ، می توان هر زیر سیستم را به یک تیم RAD سپرد و در انتها تمام زیر سیستمها را یکپارچه نمود.

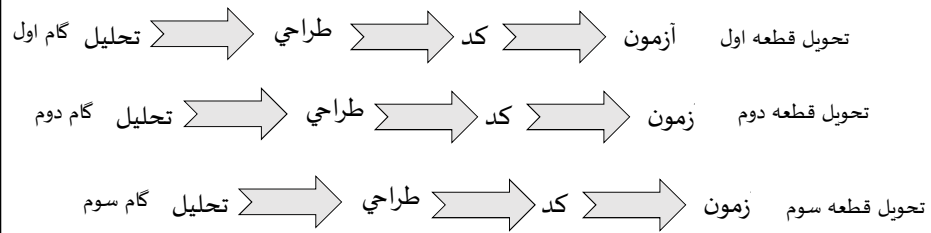
• معایب روش

مدل RAD (ادامه)



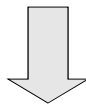
مدل افزایشی

- ترکیب مدل خطی و مدل ساخت نمونه اولیه
- در انتهای هر ترتیب خطی یک محصول از نرم افزار ارائه می گردد. اولین محصول با نام محصول هسته ای (Core Product) به نیازمندیهای پایه ای پرداخته و پس از بازنگری توسط کاربر اصلاح و بهینه می گردد.



مدل حلزونی (Spiral Model)

- ماهیت تکراری مدل ساخت نمونه اولیه + ماهیت سیستماتیک مدل ترتیبی



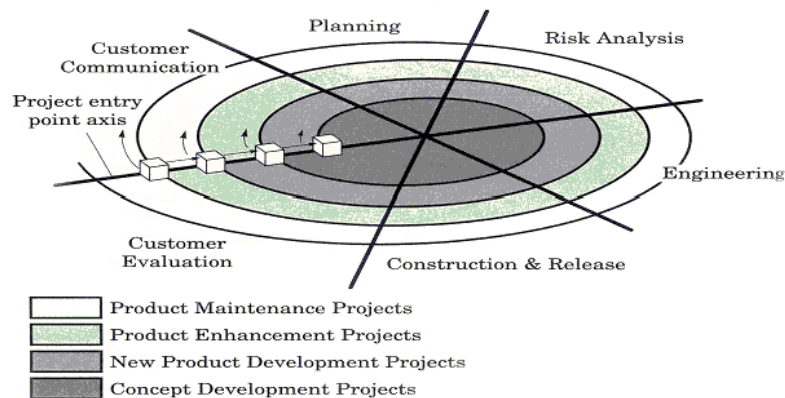
مدل حلزونی

مدل حلزونی (Spiral Model)

- نسخه اولیه از محصول در این مدل نسخه ساده ای می باشد که در تکرارهای بعدی کامل می گردد. در این مدل فعالیتها به شش دسته تقسیم می گردد که هر کدام از آنها را با نام نواحی کاری (Work Area) می شناسند:
 - تعامل با مشتری و تعریف نیازمندیها : تعیین خواسته ها از جانب مشتری
 - برنامه ریزی (Planning) : تعیین اهداف ، آلترناتیوها و محدودیتها تعیین منابع و ایجاد زمانبندی
 - آنالیز ریسک : تحلیل آلترناتیوها ، شناسایی ریسکها و راهکارهای مقابله با آنها
 - مهندسی (Engineering) : توسعه محصول سطح بعدی
 - ساخت و ارائه ، آزمایش و انتقال (تحویل مستندات ، آموزش و ...)
 - ارزیابی مشتری (Customer Evaluation) : ارزیابی نتایج مهندسی

مدل حلزونی (ادامه)

- در تمامی مراحل فوق فعالیتهای چتری نیز به موازات اجرا می گردند.



مدل حلزونی برنده برنده (Win-Win)

- در بخش تعامل با مشتری نیازمندیها از سوی مشتری می بایست مشخص شوند. جهت این موضوع لازم است مشتری به یک موازنه (trade off) بین نیازمندیهای خود و تیم توسعه برسد. به عبارت دیگر موازنه ای بین عملکرد، قابلیت‌های سیستم و کارایی از طرفی، هزینه و زمان از سوی دیگر برقرار نماید. در این شرایط تلاش می گردد اکثر نیازمندیهای مشتری در مقابل زمان و قیمت مناسب جهت تیم توسعه دهنده نرم افزار فراهم گردد (برد-برد). در مدل مذکور به جای بخش تعامل با مشتری و تعیین نیازمندیها قسمتهای زیر جایگزین می گردد:
 - شناسائی واگذارنده و تعیین شرایط برد او
 - مذاکره جهت حصول به توافق (در راستای قاعده برد - برد)

مدل روشهای رسمی (Formal Method)

- این مدل شامل مجموعه ای از فعالیتها می باشد که نرم افزار را بصورت ریاضی و رسمی بیان می نماید. به دلیل اینکه در روشهای قراردادی از تئوری مجموعه ها، نشانه گذاری منطقی و نماد گذاری ریاضی استفاده می شود، در نتیجه این روش بطور ذاتی ابهام کمتری نسبت به روشهای غیر قراردادی دارد.
- معایب:
 - زمان و هزینه بالا
 - آموزش گسترده جهت مهندسين نرم افزار
 - دشواری بکارگیری مدلها در تعامل با مشتریانی که فاقد دید فنی می باشند.

تکنیک‌های نسل چهارم (4GT)

- محور اصلی این روش استفاده از ابزارهای مهندسی نرم افزار است. مهندس نرم افزار یک مشخصه از نرم افزار را در سطح بالا تعریف نموده (معمولا با استفاده از نمادها و مدلها) سپس این ابزار کد لازم را تولید می نمایند.
- با تلفیق این تکنیک و روش "توسعه مبتنی بر مولفه" می توان به توانائی بالائی جهت توسعه نرم افزار دست یافت.

فرآیند RUP

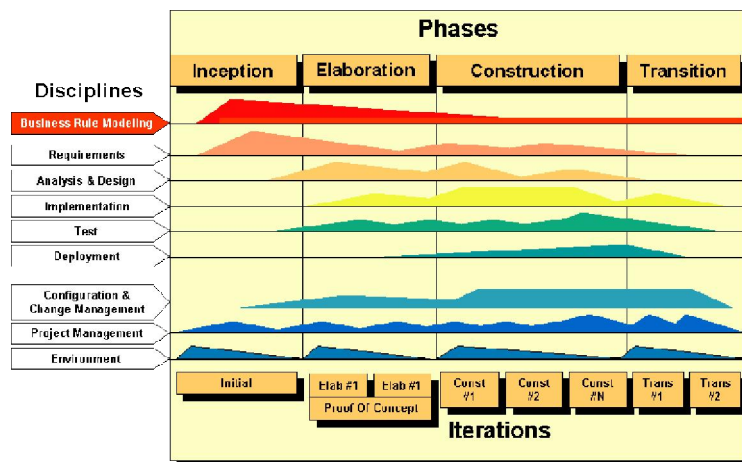
Rational Unified Process

- هدف از این فرآیند ساخت نرم افزار است به صورتیکه نیازهای کاربران را رفع نماید و از نظر هزینه و زمانبندی قابل پیش بینی باشد.
- این فرآیند دارای چهار فاز زیر می باشد:
 - ادراک
 - تشریح
 - ساخت
 - انتقال

فرآیند RUP

- فرآیند RUP شامل سه جنبه اصلی می باشد:
- دیدگاه پویا که فازهای فرآیند را در طی زمان نشان می دهد.
- دیدگاه ایستا که فعالیتهای موجود در این فرآیند را نشان می دهد.
- دیدگاه هدف گرا که عملیاتی مفید را جهت توسعه نرم افزار پیشنهاد می دهد.

فرآیند RUP



موارد سختی تولید نرم افزار

- ارتباطات
- طبیعت
- ویژگیهای خاص پروژه
- اندازه پروژه
- قابلیتها/ حجم داده های پردازش
- پیچیدگی
- بدیع بودن مسئله
- امنیت
- ویژگی های پرسنل
- کارگروهی
- دانش فرد
- مهارت در برقراری ارتباط
- مسائل مدیریتی