

بنام خدا

مهندسی نرم افزار 1

جلسه اول

مدرس: فردین شاپوری

مهندسی نرم افزار 1 - فردین شاپوری

مراجع:

- 1- R. pressman, "Software engineering: a practitioner's approach", 5th edition, McGraw-Hill, 2000.
- 2-A. Sommerville, "Software Engineering", 6th edition, Addison-Wesley, 2000.
- 3- J. Whitten, L. Bentley, and. K. C. Dittman, "Systems Analysis and Design Methods", 5th Edition, McGraw-Hill, 2001.
- 4- Yourdon, Modern Structured Analysis, Prentice-Hall, 1989.
- 5- Bently, Barlow and toppan, Systems Analysis and design methods, 1990.
- 6- فیلیپ. ال. ویور، "SSADM version 4 روش تحلیل و طراحی ساخت یافته سیستم ها"، ترجمه شرکت همکاران سیستم.
- 7- س. پارسا، "تحلیل و طراحی سیستمها (مهندسی نرم افزار)"، جلد اول، انتشارات علم و صنعت

■ تعریف نرم افزار: به مجموعه برنامه، ساختار داده ای و مستندات نرم افزار گفته می شود.

■ تعاریف مهندسی نرم افزار:

- 1- مهندسی نرم افزار عبارتست از کاربرد عملی علم کامپیوتر، علم مدیریت و سایر علوم برای تحلیل، طراحی، ساخت، نگهداری نرم افزار و مستندسازی مورد نیاز برای استفاده در اجرا و نگهداری نرم افزار تولید شده
- 2- مهندسی نرم افزار عبارتست از کاربرد اصول مهندسی برای بهینه سازی روند تولید و نگهداری سیستم های نرم افزاری با هدف کاهش هزینه و افزایش قابلیت اطمینان
- 3- تعریف IEEE: استفاده از روشهای سیستماتیک، با قاعده و قابل اندازه گیری برای تولید، بکارگیری و نگهداری نرم افزار
- 4- تعریف و استفاده از قواعد و اصول صحیح مهندسی است برای دستیابی به نرم افزار بصورت اقتصادی که دارای قابلیت اطمینان بوده و بصورت کارا روی ماشین های واقعی کار کند.

طبقه بندی نرم افزارها:

- 1- الف) batch : اطلاعات را در یک فایل بنویسیم و بدهیم
ب) online : اطلاعات از طریق واسط یا رابط کاربر وارد شود و کاربر ارتباط مستقیم با نرم افزار در حال اجرا داشته باشد.
- 2- الف) intelligent: محاسبات غیر عددی، مثل قوانین وکالت،
ب) non-intelligent: محاسبات عددی
- 3- الف) event-driven: مثل ویژوال ها
ب) non-event-driven

4- طبقه بندی بر اساس کاربرد:

- نرم افزارهای سیستمی مانند OS، editors، linkers، ...
- نرم افزارهای بلادرنگ: نرم افزارهایی هستند که وقایع دنیای واقعی را اندازه گیری، تجزیه و تحلیل و کنترل می کنند، مانند سیستم های هواشناسی، ابزار جنگی در کشتی ها، پالایشگاه ها، ...
- بخش های مختلف این گونه نرم افزارها عبارتند از:
 - بخش جمع آوری کننده اطلاعات از محیط
 - عنصر تحلیل گر
 - عنصر کنترل گر که به محیط خارجی پاسخ می دهد
 - بخش هماهنگ کننده
- نرم افزارهای تجاری مانند حسابداری، انبارداری، کتابخانه، ...
- نرم افزارهای توکار (embedded software) مثل نرم افزار جا داده شده در ماشین لباسشویی
- نرم افزارهای کامپیوترهای شخصی (personal computer software) مثل word، game، ...

ادامه طبقه بندی بر اساس کاربرد:

- نرم افزارهای هوش مصنوعی مثل سیستم آموزش دهنده هوشمند، سیستم های کنترل ربات، سیستم های تشخیص پزشکی (MYCIN)، سیستم های غلط یاب منطقی، ...
- نرم افزارهای کنترل شبکه محلی یا سراسری (LAN, WAN, WEB) یا نرم افزارهای مدیریت شبکه (Network Management Software)

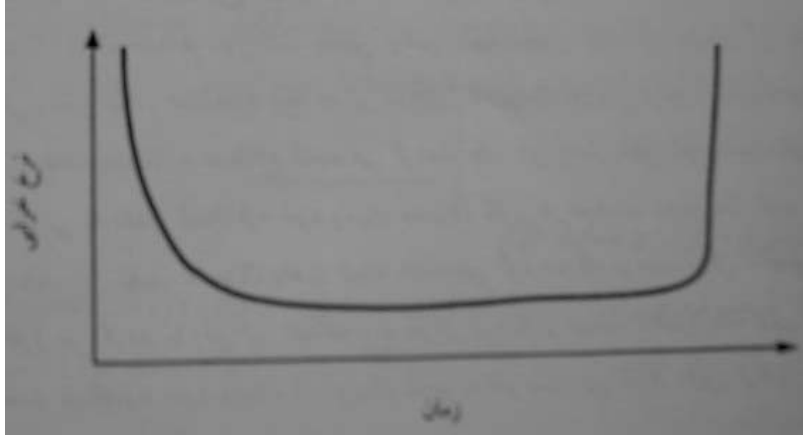
علل پیدایش بحران نرم افزار

- هزینه افزون بر بودجه
 - تاخیر در تولید و تحویل نرم افزار
 - عدم تطابق نرم افزار تحویل داده شده با مشخصات تعیین شده
 - کیفیت پایین نرم افزار
 - نگهداری پر هزینه نرم افزار
 - افزایش کاربردهای کامپیوتر
 - افزایش پیچیدگی کاربردها
 - کمبود نیروی انسانی متخصص
 - افزایش قدرت روزافزون سخت افزار
- * با توجه به این مشکلات، در سال 1967 بحث بحران نرم افزار مطرح شد و به دنبال آن با برگزاری کنفرانس هایی جهت رفع این بحران علم مهندسی نرم افزار ظهور کرد و تا به امروز در حال تکامل می باشد.

تفاوت سخت افزار با نرم افزار

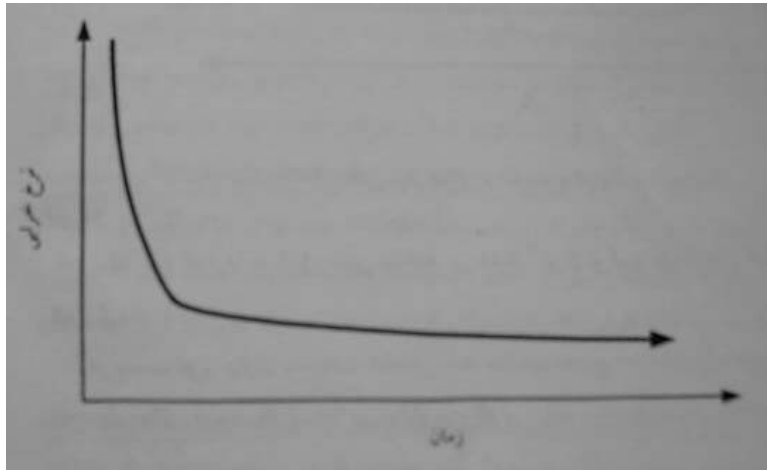
- سخت افزار یک مفهوم فیزیکی است و می توان آن را لمس کرد. پیشرفت در تولید سخت افزار قابل مشاهده است. لیست نیازمندیها و خواسته ها جهت ساخت یک سخت افزار توسط متخصص تعیین می شود و پس از تعیین معمولاً در طی ساخت تغییر نمی کند.
- نرم افزار یک مفهوم منطقی است تا فیزیکی. قابل لمس نیست. پیشرفت را در مهندسی نرم افزار با استفاده از مستندات می سنجیم. لیست نیازمندیها و خواسته ها معمولاً توسط کارفرما و کاربران مشخص می شود و در طی فرایند تولید و توسعه می تواند تغییر کند.

منحنی نرخ خرابی سخت افزار نسبت به نرم افزار



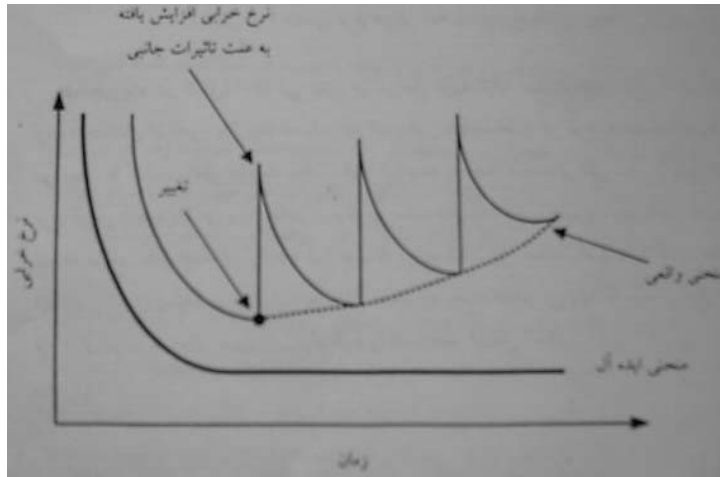
9

منحنی نرخ خرابی ایده ال نرم افزار نسبت به زمان



10

منحنی نرخ خرابی واقعی نرم افزار نسبت به زمان



11

■ تعریف سیستم (نظام): یک مجموعه از مولفه هایی که برای انجام دادن یک کار و یا کنترل یک فرایند سازماندهی شده باشند.

■ تولید یک نرم افزار بطور کلی دارای سه فاز است:

1- تعریف

2- تولید

3- نگهداری

■ در فاز تعریف بررسی های زیر انجام می شود:

1- چه اطلاعات و داده هایی باید پردازش شود.

2- چه عملیات و کارهایی لازم است.

3- چه واسطه هایی (نرم افزاری یا سخت افزاری) لازم است.

4- چه محدودیت هایی وجود دارد.

5- ضوابط ارزیابی سیستم باید مشخص شود.

* در این فاز بر روی اینکه چه چیزی مورد نیاز است، تاکید می شود.

12

* به طور کلی در فاز تعریف سه قدم (گام) اصلی وجود دارد:

1- تجزیه و تحلیل سیستم (system engineering): در این گام نقش هر مولفه در سیستم (سیستمی که قرار است تولید شود) و نقش کل سیستم نیز تعیین می شود. به عبارت دیگر حوزه عمل سیستم (scope یا domain) و بستر لازم چه سخت افزاری و چه نرم افزاری تعیین می شود. به علاوه بررسی کلان در مورد کل سازمان و کارهای مربوطه در این گام انجام می گیرد.

2- برنامه ریزی (planning) پروژه انجام می شود. در این گام، برنامه ریزی انسانی و هزینه و زمان و زمانبندی انجام کارها مشخص می شود.

3- تجزیه و تحلیل نیازهای سیستم مورد نظر. در این مرحله عملیات مورد انتظار از سیستم تعیین و پرونده های (بانکهای) اطلاعاتی لازم تعریف و سپس فرمهای ورودی و خروجی مشخص می شود.
گام سوم به یک تکه از کل نگاه می کند ولی گام یا قدم اول کل سیستم را می بیند که در خیلی از پروژه ها، متأسفانه به کل توجهی نمی شود. در نتیجه سیستم بصورت وصله وصله در می آید نه یک تکه.

13

■ فاز تولید: در این فاز بر روی چگونگی تاکید می شود و چگونگی تبدیل مرحله تحلیل (گام 3) به نرم افزار (برنامه ها) مورد توجه قرار می گیرد.

- ساختار داده ها و معماری نرم افزار تعیین می شود.
- جزئیات رویه ها (process) تعیین می شود.
- چگونگی تبدیل طراحی به برنامه کامپیوتری مشخص می شود.
- ایجاد داده های آزمایشی برای تست نرم افزار

■ سه گام اصلی در این فاز وجود دارد:

- 1- طراحی نرم افزار: در این گام نیازمندیهای نرم افزار به صورت مجموعه ای از نمایشها (مانند گرافیکی، جدولی و غیره) که ساختمان داده ها و معماری و الگوریتم ها را توضیح می دهد، تبدیل می شود.
- 2- تولید کد: در این گام، نتایج طراحی با کمک یک زبان برنامه سازی به برنامه ها تبدیل می شود. معمولاً از 4GL یا 3GL استفاده می شود. البته این به پارامترهای بسیاری از جمله موقعیت مکانی و زمانی سازمان، تکنولوژی فعلی سازمان، روش طراحی (مثلاً شنی گرا) و ... بستگی دارد.
- 3- تست نرم افزار: در این گام، نرم افزار تولید شده را جهت بر آورده شدن نیازهای (خواسته های) مشتری تست می کنیم. در این راستا unit test (تست تک تک برنامه ها) بعد module test (تست پیمانه یا بخش) و نهایتاً system test یا تست کامل نرم افزار صورت می گیرد.

14

■ فاز تعمیر و نگهداری (maintenance): در این فاز روی تغییراتی که پس از تحویل به مشتری پیش می آید تاکید می شود. این تغییرات عبارتند از:

1- تعمیرات اصلاحی (corrective maintenance)

2- تعمیرات تطبیقی (adaptive maintenance) به عنوان مثال سیستم عامل تغییر کرد یا زبان برنامه سازی جدید آمد و یا تغییراتی سخت افزاری اعمال گردید یا در قوانین سازمانی تغییراتی داده شد.

3- تعمیرات تکمیلی (perfective maintenance) به عنوان مثال دانشجویان شبانه هم اضافه شود و ...