# تست نرم‌افزار

Azimi.marjan@gmail.com

# منبع درس:

- مهندسی
  آزمون، اعتبارسنجی و تست نرم‌افزار

تالیف و تدوین: ناصر مدیری-رویا واعظی

# ارزشیابی

| | |
|---|---|
| امتحان میان‌ترم | ۳۰٪ |
| امتحان پایان‌ترم | ۳۵٪ |
| تمرین | ۵٪ |
| ارایه | ۲۰٪ |
| امتحانک و فعالیت کلاسی | ۱۰٪ |

# Verification & Validation

- وارسی: فرآیندی است که برای اطمینان از تطابق ویژگی‌های تولیدات یک فعالیت در چرخه توسعه نرم‌افزار، با نیازهای اعلام شده همان مرحله انجام می‌شود.

- اعتبار سنجی: یعنی بررسی این که خروجی مطابق با خواسته‌ها باشد. اعتبارسنجی فرآیند مقایسه مشخصات محصول با نیازمندی‌های واقعی کاربر است.
  - اعتبار سنجی بنا به تعریف IEEE عبارت است از ارزیابی نرم‌افزار در پایان تولید به منظور حصول اطمینان از رعایت نیازهای کاربر. بنابراین اعتبارسنجی، وارسی نهایی است.

# تفاوت وارسی و اعتبار سنجی

| وارسی | اعتبارسنجی |
| --- | --- |
| آیا محصول نرم‌افزاری به درستی تولید می‌شود؟ | آیا محصول درستی تولید می‌شود؟ |
| بررسی تطابق نرم‌افزار با مشخصات تعیین شده | بررسی تطابق نرم‌افزار با انتظارات مشتری و نیازمندی‌ها |
| تمرین در سطح پایین | تمرین در سطح بالا |
| به طور کلی برای اولین بار قبل از اعتبار سنجی انجام می‌شود | به طور کلی پس از وارسی انجام می‌شود |
| استفاده از روش‌هایی مانند بازدید، بررسی، تکمیل فرم سفارش، چک کردن و ... | استفاده از روش‌هایی مانند تست جعبه سیاه، جعبه سفید و .... |

# V&V

- V&v شامل بسیاری از فعالیت‌هایی است که تضمین کیفیت نرم‌افزار نامیده می‌شود. شامل: مرورهای فنی رسمی، بررسی کیفیت و پیکربندی، نظارت بر کارایی، شبیه‌سازی، امکان‌سنجی، مرور مستندات، مرور بانک اطلاعاتی، تحلیل الگوریتم و بسیاری فعالیت‌های دیگر.

- هدف اصلی در فرآیند نرم‌افزار v&v تعیین این است که نرم‌افزار توابع در نظر گرفته شده را به درستی انجام می‌دهد. همچنین اطمینان حاصل می‌شود که هیچ تابع ناخواسته‌ای اجرا نخواهد شد. روند V&Vشدیدا با فرآیند توسعه نرم‌افزار یکی شده است.

# دو رویکرد اساسی وارسی

- وارسی پویا: به عنوان تست یا آزمایش شناخته می‌شود و برای یافتن نقص مفید است.

- وارسی ایستا: به عنوان تجزیه و تحلیل شناخته می‌شود و برای اثبات صحت یک برنامه بسیار مفید است.

# وارسی پویا

- در طول اجرای نرم‌افزار انجام می‌شود که عموما به عنوان مرحله آزمایش شناخته می‌شود. بسته به دامنه آزمایش به سه دسته زیر دسته‌بندی می‌شود:
  - تست کوچک: چک کردن یک تابع یا کلاس که تست واحد نامیده می‌شود.
  - تست بزرگ: چک کردن گروهی از کلاس‌ها می‌باشد، از جمله:
    - تست ماژول(یک ماژول منفرد)
    - تست یکپارچگی(بیش از یک ماژول)
    - تست سیستم(کل سیستم)
  - تست پذیرش(تست رسمی تعریف شده برای بررسی معیارهای پذیرش یک نرم‌افزار)

# وارسی ایستا

- با انجام وارسی فیزیکی چک می‌کند که آیا نرم‌افزار مطابق با نیازمندی‌های مورد نیاز است.
  - متریک‌های نرم‌افزاری انجام محاسبات
  - وارسی رسمی
  - شیوه‌های بد تشخیص
  - ...

# وارسی و اعتبارسنجی در چرخه حیات تولید نرم‌افزار

- وارسی با تکیه بر روش‌های بازدید و غیر اجرایی تضمین می‌کند که این سیستم با استانداردها و فرآیندهای سازمان مطابقت دارد. اعتبارسنجی توسط اجرای عملکرد سیستم از طریق مجموعه‌ای از تست‌ها که می‌توانند مشاهده و ارزیابی شوند، از لحاظ فیزیکی تضمین می‌کند که سیستم مطابق با طرح عمل می‌کند.

# فعالیت‌های وارسی در چرخه نرم‌افزار

| مراحل چرخه نرم‌افزار | فعالیت‌های وارسی |
| --- | --- |
| نیازمندی‌ها | تعیین رویکرد وارسی<br>تعیین کفایت نیازمندی‌ها<br>تولید داده‌های تست عملکردی<br>تعیین سازگاری طراحی با نیازمندی‌ها |
| طراحی | تعیین کیفیت طراحی<br>تولید داده‌ها تست کاربردی و ساختاری<br>تعیین سازگاری با طراحی |
| ساخت | تعیین کیفیت پیاده‌سازی<br>تولید داده‌های کاربردی و ساختاری برای برنامه<br>اعمال داده‌های تست |
| بهره‌برداری و نگهداری | وارسی مجدد، متناسب با سطح توسعه مجدد |

# روش‌های وارسی

- خود بازدیدی: این اساسا یک فرآیند خود یادگیری است. خود بازدیدی نسبت به زمان و یافتن نقص بسیار انعطاف پذیر است و به عنوان یک نیاز، قرار ملاقاتی برای آن وجود ندارد. نقص یافت شده در خود بازدیدی می‌تواند در خود بهبودی کمک کند.

- بازدید قرین(نظیر): یک قرین ممکن است یک شخص توسعه دهنده یا تست کننده باشد. بازدید قرین بارها در فازهای مختلف چرخه حیات تولید نرم‌افزار انجام شده است. دو نوع بازدید وجود دارد:
  - بازدید آنلاین: نویسنده و بازبین‌گر با هم ملاقات می‌کنند و محصول کار را به طور مشترک بازدید می‌کنند.
  - بازدید آفلاین: نویسنده بازبین‌گر را مطلع می‌کند که محصول آماده است و بازبین‌گر ممکن است محصول را با توجه به زمان در دسترس خود مورد بازدید قرار دهد.

# روش‌های وارسی

- جلسه بررسی: یک نوع نیمه رسمی از بازدید است که تیم بزرگتری به همراه نویسنده کار را بازدید می‌کنند. جلسه بررسی به طور موثر برای ارتباط بین تیم استفاده می‌شود. حاصل یک جلسه بازبینی می‌تواند تایید محصول یا فرآیند، تعیین اشکال یا اشکالات و پیشنهادهای جایگزینی دیگر باشد.
  - قبل از تشکیل جلسه:
    - تعیین موضوع، اعضا، زمان و مکان برگزاری جلسه
    - دعوت از اعضای جلسه به همراه ارسال موضوع و دستور کار جلسه
  - در حین تشکیل جلسه:
    - توجیه همه اعضا نسبت به موضوع و دستور کار جلسه
    - ثبت مذاکرات و مصوبات جلسه به صورت رسمی
  - بعد از تشکیل جلسه:
    - تهیه گزارش رسمی تشکیل جلسه
    - ارسال نتایج جلسه به افراد و گروه‌های ذینفع

# روش‌های وارسی

- بازرسی(بازدید رسمی): این یک بازدید رسمی است که در آن افراد خارجی به عنوان بازرس درگیر هستند. آن‌ها کارشناس موضوع هستند که محصول کار را بازدید می‌کنند. نواقص ثبت می‌شوند اما راه‌حل‌ها توسط کارشناسان موضوع معین نمی‌شوند. این کمک می‌کند تا سازمان اقدام خود برای رفع نواقص را آغاز کند.

- ممیزی: یک بررسی رسمی بر اساس نمونه‌هاست. ممیزی توسط مامور رسیدگی که ممکن است یا ممکن نیست در محصول کار داده شده کارشناس باشد انجام می‌شود. فعالیت‌های اعتبارسنجی نرم‌افزار ممکن است در این مدت و همچنین در انتهای چرخه حیات نرم‌افزار برای اطمینان از این که تمام نیازمندی‌ها برآورده شده است، انجام شود.

# روش‌های اعتبارسنجی

| نمونه اعتبار سنجی | انجام شده توسط | توضیح |
|---|---|---|
| تست واحد | توسعه دهندگان | تست یک برنامه واحد، ماژول یا واحد کد |
| تست یکپارچگی | توسعه دهندگان با پشتیبانی از یک تیم مستقل تست | تست یک برنامه واحد، ماژول یا واحد کد. تعیین اعتبار این که بخش‌های مختلف سیستم متقابلا بر طبق طراحی سیستم هستند. |
| تست سیستم | تیم مستقل تست | تست از کل سیستم کامپیوتری است. این نوع از تست می‌تواند شامل تست عملکردی و ساختاری باشد. |
| تست پذیرش کاربر | کاربر با پشتیبانی تیم مستقل تست | تست سیستم کامپیوتری یا بخش‌هایی از سیستم کامپیوتری برای اطمینان از این که در سیستم کار خواهد کرد، صرف نظر از این که نیازمندی‌های نرم‌افزار چه چیزی را بیان می‌کنند. |

# نقش وارسی و اعتبار سنجی

- نقش V&V باید برای هر محصول به صورت پروژه به پروژه تعیین شود. این تصمیم توسط حساسیت محصول، محدودیت‌های آن و پیچیدگی آن تحت تاثیر قرار می‌گیرد. به طور کلی هدف از تابع V&V اطمینان حاصل کردن از این است که محصول نیازمندی‌های کاربر را ارضا می‌کند.

# استاندارد طرح وارسی و اعتبار سنجی

- فرآیندهای وارسی و اعتبار سنجی از مهم‌ترین فرآیندهای پشتیبان توسعه نرم‌افزار است که اجرای صحیح و دقیق آن‌ها نقش عمده‌ای در تضمین کیفیت پروژه‌های نرم‌افزاری ایفا می‌کند. گام نخست در اجرای این فرآیندها، برنامه ریزی فعالیت‌های V&V است که نتیجه آن به شکل طرح وارسی و اعتبار سنجی تدوین و ارایه می‌گردد. از این رو استاندارد سازی قالب و محتوای اینگونه طرح‌ها به منظور ارتقای کیفی پروژه‌های نرم‌افزاری، اقدامی ضروری است.

هدف از تهیه این استاندارد، یکسان‌سازی طرح‌های وارسی و اعتبارسنجی در پروژه‌های نرم‌افزاری و فراهم آوردن امکان ممیزی و کنترل کیفیت این گونه طرح‌هاست.

# مدیریت فرآیندهای V&V

- **سازمان:** عناصر درگیر در انجام فعالیت‌های V&V همراه با شرح وظایف مربوطه در این بند معرفی می‌شوند. واحدها و عناصر باید در قالب یک نمودار تشکیلاتی تشریح گردند. ضوابط ترسیم این نمودار به استانداردهای سازمانی کارگزار بستگی دارد.

- **مقاطع زمانی انجام فعالیت‌های V&V:** در این بند باید مقاطع زمانی انجام هر یک از فعالیت‌های V&V پروژه تعیین شود. مقاطع زمانی فعالیت‌های وارسی عموما در سند متدودولوژی پروژه و در کنار سایر فعالیت‌های فنی پروژه مشخص گردد. زمان انجام فعالیت‌های اعتبارسنجی معمولا در پایان مراحل تولید نرم‌افزار می‌باشد.

- **زمانبندی انجام فعالیت‌های V&V:** در این بند باید زمانبندی انجام فعالیت‌های V&V بر اساس زمانبندی ارایه شده در طرح مدیریت پروژه باشد که در آن فعالیت‌های V&V به نحوی متمایز شده‌اند.

# مدیریت فرآیندهای V&V

- **منابع مورد نیاز:** منابع مورد نیاز برای انجام فعالیت‌های V&V باید در این بند تعیین گردد.
  - نیروی انسانی
  - سخت افزار
  - نرم افزار
  - سایر منابع مورد نیاز
- **ابزارها و روش‌ها:** در این بند ابزارها و روش‌های به کار گرفته شده برای انجام فرآیندهای V&V مانند چک لیست‌ها، ابزارهای نرم افزاری و ... و الزامات استفاده از آن‌ها باید مشخص شود.

# فعالیت‌های V&V

- فعالیت‌های طراحی شده برای وارسی و اعتبار سنجی، به تفکیک مراحل چرخه حیات توسعه،همچنین فعالیت‌های لازم برای مدیریت فرآیندهای V&V در این جا باید ارایه گردد. به ازای هر فعالیت مشخصات زیر باید ارایه گردد:
  - عنوان فعالیت
  - شناسه
  - هدف
  - روش‌ها و معیارها
  - ورودی‌ها
  - فرآورده‌ها

# گزارش‌دهی

- کلیه فعالیت‌هایی که در چارچوب فرآیندهای V&V پروژه انجام می‌شود، باید به صورت رسمی ثبت شده و نتایج آن‌ها به نحو مناسبی گزارش شود.
- گزارش فعالیت
- گزارش اشکال

# ارزیابی نرم‌افزار

- ارزیابی بسیار مهم‌تر از تست و پیاده‌سازی است. ارزیابی جهت اندازه‌گیری موفقیت در پیاده‌سازی و طرح‌های امنیتی ضروری است. ارزیابی برای مشتریان نیز بسیار مهم است. نتایج حاصل از ارزیابی می‌تواند منجر به نیازهای جدید و تغییر درخواست‌ها شود. در ارزیابی نرم‌افزار در نظر است تا نرم‌افزار از سه جنبه اصلی مورد ارزشیابی قرار گیرد. این سه فاکتور عبارتند از:
  - قابلیت استفاده
  - قابلیت تحمل
  - قابلیت نگهداری

# قابلیت استفاده(Usability)

- مجموعه‌ای از مشخصه‌هاست که مربوط به نیازهای استفاده از نرم‌افزار می‌باشد. این فاکتور وابسته به گروه کاربران نرم‌افزار است. قابلیت استفاده وابسته به این است که چه کسی از نرم‌افزار استفاده می‌کند و به تعریف کاربران از استفاده آسان بستگی دارد.
- در قابلیت استفاده بودن موارد زیر مطرح می‌شود:
  - قابل فهم بودن نرم‌افزار
  - مستند سازی
  - قابل ساخت بودن
  - قابلیت نصب شدن
  - قابلیت یادگیری

# قابلیت نگهداری

- قابلیت نگهداری دارای مجموعه‌ای از مشخصه‌هاست که مربوط به نیازهای تغییر نرم‌افزار است.

- نگهداری یعنی ارزیابی نرم‌افزار و یافتن خطاهای موجود و اصلاح آن‌ها و اطمینان از این که این تغییرات دارای اثر جانبی نبوده و کیفیت نرم‌افزار را به مخاطره نمی‌اندازد. سپس نسخه جدید مجددا مورد بررسی قرار می‌گیرد.

# قابلیت تحمل

- دارای مجموعه‌ای از مشخصه‌هاست که تعیین می‌کند که نرم‌افزار می‌تواند خود را با تغییرات سخت افزاری و نرم‌افزاری که با سرعت رخ می‌دهد وفق دهد یا خیر. به عبارت دیگر قابلیت تحمل تغییرات پیرامون خود را دارد یا خیر.

# قابلیت تحمل

- در قابلیت تحمل موارد زیر مطرح می‌شود:
  - هویت
  - حق کپی رایت
  - مجوز
  - نظارت
  - انجمن
  - قابلیت دسترسی
  - قابلیت تست
  - قابلیت حمل
  - قابلیت پشتیبانی
  - قابلیت تجزیه و تحلیل
  - قابلیت تغییر
  - قابلیت توسعه
  - قابلیت تطبیق و تعمیم

# فاکتورهای قابلیت تحمل

- هویت: در این فاکتور ارزیابی مواردی از قبیل این  که نرم‌افزار نام و لوگوی مخصوص دارد، بررسی می‌شود.
- حق کپی رایت: در این فاکتور مشخص می‌شود که تا چه حد نام نویسنده نرم‌افزار و مالک کپی رایت آن واضح و روشن است و آیا نرم‌افزار حق کپی رایت دارد یا خیر.
- مجوز: در این قسمت ارزیابی می‌شود که آیا نرم‌افزار مجوز دارد یا نرم‌افزار مجوز به رسمیت شناخته شده طرح نرم‌افزار باز را دارد.

# فاکتورهای قابلیت تحمل

- نظارت: در این بخش ارزیابی می‌شود که تا چه حد و چگونه پیشرفت پروژه مدیریت یا شفاف سازی می‌شود.
- انجمن: در این فاکتور بررسی می‌شود که تا چه حد کاربر فعال برای این محصول وجود دارد.
- قابلیت دسترسی: در این فاز ارزیابی می‌شود که تا چه حد نرم‌افزار قابل دسترس است و آیا توزیع بدون نیاز به هیچ گونه ثبت نام یا احراز هویت به وسیله پروژه قابل دسترس است یا خیر.

# فاکتورهای قابلیت تحمل

- قابل حمل بودن: در این بخش ارزیابی می شود که تا چه حد می‌توانید این نرم‌افزار را بر روی سیستم عامل‌های دیگر استفاده کنید.
- قابلیت پشتیبانی: این بخش به این که تا چه اندازه محصول در حال حاضر و در آینده پشتیبانی می‌شود؟ آیا وب سایت صفحه چگونگی پشتیبان گرفتن را دارد؟ آیا نرم‌افزار توضیح می‌دهد که چگونه می‌توان پشتیبانی را به دست آورد.
- قابلیت تجزیه و تحلیل: در این بخش  تجزیه و تحلیل انتشار منبع نرم‌افزار برای درک کاربرد معماری و فایل‌های کد منبع منحصر به فرد آن و این که آنها چگونه در اجرای معماری جایگذاری می‌شوند انجام می‌گیرد.

# فاکتورهای قابلیت تحمل

- قابلیت تغییر: در این بخش مسایلی مانند چگونگی تغییر یک نرم‌افزار در مسایل مربوط به آدرس، اصلاح قابلیت، اضافه کردن قابلیت‌های جدید و این که آیا تغییرات در مخزن کد منبع به یک لیست پستی ایمیل می‌شود و این لیست ارسال و دریافت ایمیل می‌تواند توسط هر کسی به اشتراک گذاشته شود، بررسی می‌شود.

- قابلیت توسعه: در این بخش این که محصول تا چه حدی در آینده قابل توسعه است، بررسی می‌شود.

- قابلیت تطبیق و تعمیم: در این بخش این که محصول تا چه حدی قابل تطبیق و تعمیم است؟ آیا قابل تطبیق با استانداردهای open است؟ بررسی می‌شود.

# معیارهای ارزیابی نرم‌افزار

- میزان پیشرفت نرم‌افزار
- میزان منابع انسانی و انرژی سرمایه گذاری شده برای پروژه
- هزینه تولید نرم‌افزار
- مشاهده نتایج نهایی نرم‌افزار
- گزارش‌های خطا و ایرادهای پروژه
- ثبات و درستی و کامل بودن نیازمندی‌های سیستم
- ثبات حجم و اندازه پایداری طول و عرض یک پروژه
- مقدار استفاده پروژه از منابع سخت افزاری کامپیوتر
- آموزش در نظر گرفته شده پس از پیاده سازی محصول نرم افزاری

# مدل‌های ارزیابی کیفیت نرم‌افزار

- سلسله مراتبی : ویژگی‌های کیفیت در سطح اول و دوم خصوصیات فرعی متناظر با ویژگی‌ها قرار دارند. چنانچه مدل دارای معیار اندازه‌گیری هم باشد، این معیارها در سطح سوم مدل قرار می‌گیرند. ارتباط بین اجزای سطوح یک مدل کیفیت می‌تواند یک به چند یا چند به چند باشد. در ارتباط یک به چند هر فاکتور یا خصوصیت کیفی فقط با خصوصیات فرعی خود در سطح پایین‌تر مرتبط است. ولی در ارتباط چند به چند هر فاکتور می‌تواند با خصوصیات فرعی دیگر نیز مرتبط باشد. کیفیت نرم‌افزار از ارزیابی این خصوصیات فرعی توسط معیارهای اندازه‌گیری حاصل می‌شود.

- غیر سلسله مراتبی: دارای ساختار یکسانی نیستند.

# مدل McCall

- این مدل در سال ۱۹۷۶ توسط نیروی هوایی آمریکا، جنرال الکتریک و مرکز توسعه هوایی روم با هدف بهبود کیفیت محصولات نرم‌افزاری ارایه شد.
- سطح اول مدل شامل ۱۱ خصوصیت کیفی صحت، قابلیت اطمینان، کارایی، قابلیت استفاده، قابلیت نگهداری، آزمایش پذیری، انعطاف پذیری، انتقال پذیری، قابلیت استفاده مجدد و قابلیت همکاری است.
- در سطح دوم مدل نیز ۲۳ معیار کیفی ارایه شده است که ارتباط چند به چند با ویژگی‌ها اصلی سطح اول دارد.
- ایده اصلی مدل تعیین ارتباط بین عوامل کیفی و معیارهای ارزیابی محصول است.
- مزیت عمده این مدل ارتباط بین خصوصیات کیفی و معیارهاست.

# مدل Boehm

- این مدل در سال ۱۹۷۸ برخی خصوصیات را با تاکید بر قابلیت نگهداری نرم افزار به مدلMcCall اضافه کرد.
- همچنین این مدل ملاحضاتی در خصوص ارزیابی نرم‌افزار با توجه به نوع کاربرد آن و خصوصیاتی مرتبط با سخت افزار اضافه کرد.
- عیب اصلی این مدل عدم ارایه راهکاری به منظور ارزیابی و اندازه‌گیری خصوصیات کیفی است.

# مدل FURPS

- این مدل در سال ۱۹۸۷ توسط دو شرکت HP و Robert Grady ارایه شده است.
- شامل دو گروه متفاوت از نیازمندی‌های نرم‌افزار است:
  - نیازمندی‌های عملیاتی که با ورودی و خروجی مورد نیاز تعریف می‌شود.
  - نیازمندی‌های غیر عملیاتی که شامل چهار ویژگی قابلیت استفاده، قابلیت اطمینان، کارایی و قابلیت پشتیبانی است.

- عیب این مدل عدم وجود معیاری برای سنجش میزان انتقال پذیری نرم افزار است.

# مدل Dromey

- ایده اصلی این مدل که در سال ۱۹۹۵ ارایه شد این بود که بتواند به طور وسیعی انواع سیستم‌ها را با کاربردهای مختلف پوشش دهد.
- مراحل طراحی این مدل:
  - انتخاب مجموعه‌ای از صفات سطح بالا که برای ارزیابی لازم است.
  - تهیه لیستی از اجزای سیستم
  - تشخیص خصوصیات دارای کیفیت برای هر جز سیستم
  - تصمیم راجع به این که هر خصوصیت چگونه  بر صفات کیفیت تاثیر می‌گذارد.
  - ارزیابی مدل
- این مدل به دنبال تاثیر خصوصیات محصول نرم‌افزار بر صفات کمی است.

# مدل ISO-IEC-9126

- با توجه به نیاز شدید صنعت نرم‌افزار به استاندارد شدن ارزیابی نرم‌افزار، این مدل ابتدا در سال ۱۹۹۱ توسط موسسه بین المللی ISO انتشار یافت و بعد از گذشت تقریبا یک دهه در سال ۲۰۰۱ توسط متخصصان ISO اصلاح و تکمیل شد.
- در سطح اول مدل، کیفیت محصول نرم‌افزاری را به شش ویژگی کیفی اصلی تقسیم می‌کند که هر یک از آن‌ها از چند ویژگی فرعی تشکیل شده‌اند.
- ارتباط ویژگی‌های سطح اول مدل با ۲۱ ویژگی فرعی مدل با سطح دوم، به صورت یک به چند است. به طوری که در این مدل کمترین همپوشانی وجود دارد. علاوه بر این دو سطح مدل دارای معیارهایی برای ارزیابی کیفیت نرم‌افزار می باشد.
- مهمترین مزیت این مدل این است که خصوصیات کیفی داخلی و خارجی یک نرم‌افزار در آن تفکیک شده است.

# مدل غیر سلسله مراتبی- ستاره‌ای

- مدل ستاره‌ای کیفیت نرم‌افزار یک مدل مفهومی برای نشان دادن دیدگاه‌های مختلف کیفیت نرم‌افزار است.
- اجزای اصلی تشکیل دهنده این مدل:
  - خریدار
  - تامین کننده
  - محصول
- خریدار با تامین کننده برای ساخت محصول نرم‌افزاری قراردادی دارند که این قرارداد به صورت کاملا روشن و واضح خصوصیات کیفی محصول را تعیین می‌کند.
- دیدگاه خریدار از محصول این است که باید توسط کاربران قابل قبول و قابل پذیرش باشد و توسط متخصصین تامین کننده پشتیبانی شود.