



QUALITY CHECK

# فنون آزمون نرم افزار

و راهبردهای اجرای آن

پژوه کنترل کیفیت  
و بررسی نرم افزار



## چکیده

لازمه یک محصول موفق و باکیفیت، رعایت اصول استاندارد و تعریف شده مربوط به آن است. بنابراین، می توان به صورت کلی این طور برداشت کرد که رعایت اصول در بخش تست، باعث تولید نرم افزاری با استاندارد و کیفیت مطلوب می گردد. شیوه های مختلفی در این راستا وجود دارد که هر کدام نیز مزایایی را دارند. برخی از این شیوه ها، قابلیت خودکارسازی را دارند که انتظار می رود با توسعه کنترل کیفیت، وارد این مقوله شوند و با کمترین هزینه و بالاترین دقت، محصول خروجی بگیرند. در این نوشتار، به معرفی شیوه های آزمون نرم افزار و راهبردهای اجرای آن می پردازیم.

**کلیدواژگان:** تست های نرم افزاری، اجرای تست، انواع تست، ارزیابی نرم افزار.

## شیوه‌های بررسی نرم‌افزار

شیوه‌های بررسی نرم‌افزار عبارت‌اند از:

### ۱. تست جعبه سفید (White-Box test)

نام دیگر این روش، تست جعبه شیشه‌ای است و با استفاده از ساختارهای درونی یک قطعه، حالات تست را استخراج می‌کند. در این روش، سعی می‌شود حالات تست به گونه‌ای استخراج شوند که موارد زیر تضمین گردند:

- تمامی مسیرهای کنترلی مستقل در کد یک قطعه، دستکم یکبار تست شده باشند؛
- تمامی تصمیم‌گیری‌های منطقی (if) بر اساس درست یا نادرست بودن شرط آن تست شده باشند؛
- تمامی حلقه‌های تکرار (loop) در محدوده تکرار و ابتدا و انتهای شرط حلقه تست شده باشند؛
- تمامی ساختارهای داده‌ای داخلی از جهت اعتبارسنجی واریسی شده باشند.

### ۲. تست مسیرهای بنیادی (Basic Path Testing)

در این روش، سعی می‌گردد تا تمامی مسیرهای کنترلی برنامه حداقل یکبار اجرا شوند تا بدین صورت، تمامی دستورهای برنامه اجرا شده باشند. برای انجام این تست، ساختار جریان کنترلی به صورت گراف، جریان در نظر گرفته می‌شود. این گراف جریان کنترلی، منطق درون برنامه را نشان می‌دهد و ساختار برنامه را می‌توان به صورت چنین گرافی بیان کرد. در این روش، برای استخراج حالات تست، با استفاده از طراحی و یا کد منبع گراف، جریان متناظر آن رسم می‌گردد. میزان پیچیدگی منطق درون برنامه برای گراف جریان تعیین می‌شود و برای هرکدام از مسیرهای مستقل، یک حالت تست آماده می‌گردد تا تضمین کند که آن حالت، حتماً اجرا می‌شود.

### ۳. تست ساختارهای کنترلی (control structures)

این روش، شرط‌های منطقی (if) موجود در برنامه را تست می‌کند. شرط‌ها می‌توانند به انواع مختلفی باشند:

- شرط ساده: یک متغیر بولی؛
- عبارات رابطه‌ای: عباراتی با عملگرهای رابطه‌ای  $>$  و  $>=$  و  $=$  و  $<$  و  $<=$  و  $=$ ؛
- شرط مرکب: ترکیبی از شرط‌های ساده، عملگرهای بولی و گرانتر.

### ۴. تست حلقه (Ring test)

حلقه‌ها، عناصر مهم و سازنده اغلب الگوریتم‌هایی هستند که در نرم‌افزار به کار می‌روند. برای یک حلقه ساده که حداکثر  $n$  مرتبه باید انجام شود، تست‌های زیر را می‌توان انجام داد:

- به‌طور کلی، از روی حلقه بگذریم؛
  - حلقه تنها یک مرتبه انجام گردد؛
  - حلقه دو مرتبه انجام گیرد؛
  - حلقه  $m$  مرتبه انجام گیرد ( $n > m$ )؛
  - حلقه  $n-1$  و  $n+1$  مرتبه انجام شود.
- برای تست حلقه‌های متوالی نیز می‌توان فرض کرد که هرکدام، یک حلقه مستقل هستند و از روش تست حلقه ساده استفاده کرد.

### ۵. تست جعبه سیاه (Black-Box Testing)

نام دیگر این روش، تست رفتاری است و بر روی نیازمندی‌های عملیاتی نرم‌افزار تمرکز دارد. حالات تستی که در این روش استخراج می‌گردند، باید به گونه‌ای باشند که تمامی نیازمندی‌های عملیاتی نرم‌افزار را تست کنند. روش تست جعبه سیاه، مکملی برای تست جعبه سفید است و توسط آن، یک مجموعه نیازهای متفاوتی را می‌توان تشخیص داد. توسط این تست، می‌توان دسته خطاهای زیر را تشخیص داد:

- عملیات اشتباه و یا از قلم افتاده؛
- خطاهای موجود در واسط‌ها؛
- خطاهای موجود در ساختارهای داده‌ای و دسترسی به پایگاه داده‌ای خارجی؛
- خطاهای رفتاری یا اشتباهات، از نظر کارایی؛
- اشتباه در مقداردهی اولیه و یا کارهای اختتامی.

### تقسیم‌بندی به گروه‌های هم‌ارز

یکی از روش‌های جعبه سیاه، تست تقسیم‌بندی به گروه‌های هم‌ارز است. در این روش، دامنه داده‌های ورودی برنامه به دسته‌های متفاوتی تقسیم می‌شوند و حالات تست، بر اساس این دسته‌ها انتخاب می‌شوند. هدف اصلی استفاده از این روش، کم کردن تعداد حالات تست است؛ بدین صورت که از یک مجموعه حالات یکسان، تنها یک حالت تست اعمال شود.

## تحلیل مقادیر مرزی

معمول تعداد بسیاری از خطاها، در حالات مرزی یک ورودی رخ می‌دهد. بنابراین، این روش نیز بر این تمرکز دارد که حالات تست، حتماً شامل مقادیر مرزی باشند. این روش، مکمل روش تقسیم‌بندی به گروه‌های هم‌ارز است و حالات تستی که توسط آن استخراج می‌گردند، شامل مقادیر مرزی هر دسته هست.

## تست تشابه

در بعضی از شرایط، قابلیت اعتماد نرم‌افزار تولیدشده بسیار حساس و مورد نیاز است؛ همانند سیستم‌های پرواز و سیستم ترمز اتومبیل. در این حالت، دو تیم توسعه مختلف یک تعریف، مشخصات یکسانی را از سیستم که در مرحله تحلیل و طراحی ایجاد گردیده است، پیاده‌سازی می‌کنند. در این حال، هر دو نسخه متفاوت چک می‌گردند تا به ازای حالات تست متفاوت، خروجی‌های صحیح یکسانی را تولید کنند. سپس، هر دو سیستم به صورت موازی اجرا می‌شوند تا خروجی‌های آنها به ازای حالات متفاوت تست شود. اگر به ازای یک ورودی، یکی از نسخه‌ها خروجی متفاوت با دیگری تولید کرد، آنگاه دو نسخه برای وجود خطا تست می‌شوند.

## آزمون‌های بررسی نرم‌افزار

ابزارهایی که فرآیند تولید نرم‌افزار را بهینه می‌کنند، با استانداردهای خاص خود درباره چگونگی تولید نرم‌افزار و عملیاتی است که درون PLC باید اتفاق بیفتد تا به حد مطلوب برسد. روش‌های مختلف ارزیابی، برای آن است تا چیدمان غیربهینه و تخصص و ابزار نامناسب به کار برده نشود. ابتدا باید PLC وجود داشته باشد و بعد متریک‌های نرم‌افزاری را روی آن اعمال کنند که در این مسیر، یکسری شاخص در مراحل مختلف تعریف می‌شود.

برای بهینه‌کردن پردازش، حتماً باید روش اندازه‌گیری داشته باشیم. متریک‌ها در یک محیط قابل مدیریت، اعداد کمی هستند که با استفاده از آنها کنترل و ارتقا کیفی محصول را در راستای اهداف خواهد داشت. هدف از متریک‌های نرم‌افزاری افزایش کارایی، کیفیت و قدرت تولید تیم توسعه است که در امتداد آن، اهداف تجاری هم باید مد نظر باشد.

وقتی نسخه‌های جدید را تولید می‌کنند، استاندارد باید رعایت شود تا از نظر کیفی از بار پیش پایین‌تر نرود.

## خواص اصلی متریک‌های نرم‌افزاری

این خواص عبارت‌اند از:

۱. بالابردن کیفیت محصول؛
۲. بهترکردن عملکرد؛
۳. بهترکردن تولید گروهی.

در تصمیم‌گیری کنترل کیفیت، برای تأیید، رد یا نمره‌دهی به محصولات، از این متریک‌ها استفاده می‌شود که برخی از آنها در بخش فنی محاسبه می‌شود و در اختیار کنترل کیفیت قرار می‌گیرد و بخشی از آنها نیز در بررسی نرم‌افزار محاسبه می‌شود و در اختیار کنترل کیفیت قرار خواهد گرفت.

مدل‌های مختلفی برای استانداردکردن متریک‌ها وجود دارد؛ مثل مدل: Furps, Boehm, McCall, ajile, Dromey و مدل ISO 9126.

## تست‌های خودکار

در مرحله تست، نرم‌افزار پیاده‌سازی شده به آزمایش گذاشته می‌شود و هدف از آن، پیداکردن خطاهای سیستم است. تست در دو زمینه صورت می‌پذیرد:

اولاً، تست در مورد اجزا به طور مستقل انجام می‌گیرد که در این خصوص، هم در مورد عملیات و دستورات داخلی برنامه آزمایش می‌شود که درست عمل کند و هم در مورد وظایف خارجی پیمانها که مشخص می‌شود بر اساس ورودی‌های مختلف، آیا نتایج موردنظر به دست می‌آیند یا نه.

ثانیاً، تست در مورد مجموعه اجزا در ارتباط با هم صورت می‌گیرد که در این حالت، مشخص می‌شود آیا خواسته‌های موردنظر کاربر حاصل شده است یا نه و بدین صورت، کل نرم‌افزار مورد آزمایش واقع می‌شود.

در صورت وجود اشکال‌هایی که نتوان در خود این مرحله رفع نمود، بسته به نوع اشکال، باید به یکی از مراحل پیش، یعنی طراحی یا تعریف خواسته‌ها و یا فهم و تحلیل خواسته‌ها بازگشت نمود. خودکارسازی تست، در واقع، انجام تست‌های نرم‌افزاری است که به وسیله ماشین انجام می‌شود؛ یعنی تست‌هایی را که به صورت دستی انجام می‌شود، در هر پروژه معین کرده و به صورت خودکار، برنامه انجام تست را اجرا می‌کنیم.

گفتنی است که همه تست‌های انجام‌شده در یک نرم‌افزار قابلیت انجام به صورت ماشینی را دارند.

برای جلوگیری از هزینه‌های مالی و زمانی، معمولاً

تست‌هایی به صورت ماشینی و خودکار انجام می‌شود که در طول تولید نرم‌افزار، چندین بار انجام شود که اصطلاحاً به این‌گونه تست‌ها، تست‌های تکرارپذیر می‌گویند. تست‌های تکرارپذیر، ارزش این را دارند که به صورت ماشینی انجام شوند.

به طور معمول، برای خودکارسازی تست‌های یک نرم‌افزار مراحل ذیل انجام می‌شود:

۱. شناسایی تست‌های تکرارپذیر؛
  ۲. روال مند و قانون‌مند کردن تست‌های تکرارپذیر؛
  ۳. تقسیم کردن یک تست تکرارپذیر به چندین تست تکرارپذیر کوچک‌تر برای ارزیابی‌های دقیق؛
  ۴. پیاده‌سازی تست‌های تکرارپذیر با کمک نرم‌افزارهای انجام تست خودکار و یا نوشتن تست خودکار مربوطه؛
  ۵. تست‌های تکرارپذیر می‌تواند تحت یک زمان‌بندی خاص و یا در موقع نیاز انجام شود؛
  ۶. برخی از تست‌های تکرارپذیر با ابزارها و نرم‌افزارهای موجود قابل پیاده‌سازی نمی‌باشند که می‌توان این تست‌های را یا به صورت دستی انجام داد و در صورتی که از اهمیت بالایی برخوردار باشد، به صورت ماشینی انجام شود و چنانچه برای این کار هزینه زمانی و مالی وجود داشته باشد، می‌توان ابزاری برای انجام این‌گونه تست‌ها نوشت تا به صورت ماشینی انجام شود.
- به طور کلی، تست‌های تکرارپذیر را می‌توان در دو بخش دسته‌بندی کرد:
۱. تست‌های تکرارپذیر جعبه سفید: که توسط برنامه‌نویسان

انجام می‌گیرد و برای انجام این‌گونه تست‌ها، بیشتر از ابزارهای تست داخلی استفاده می‌شود.

۲. تست‌های تکرارپذیر جعبه سیاه: که توسط کارشناسان تست محصول انجام می‌شود و برای انجام این‌گونه تست‌ها، ابزارهای بسیاری وجود دارد؛ مثل تست: کاملیت، انی ور، سلنیم و رونکس.

از جمله مزایا و فواید تست خودکار در جعبه سفید و جعبه سیاه، این است که هزینه‌های زمانی، مالی و سرمایه انسانی را کم می‌کند.

### تفاوت تست دستی و ماشینی

در تست‌های تکرارپذیر، انجام دستی تست‌ها ممکن است به جهت خطای انسانی، برخی تست‌ها انجام نشود؛ ولی در تست ماشینی تمامی تست‌ها بدون کم و زیاد انجام خواهد شد.

در تست‌های دستی ممکن است تست مورد نظر در زمان مورد نظر انجام نشود؛ ولی تست ماشینی با زمان‌بندی دقیق در ساعت مورد نظر واقع می‌شود.

در تست ماشینی، همه تست‌ها و اتفاقات، اعم از پاس شدن تست یا فیلد شدن تست، همگی ذخیره می‌شود و تاریخچه‌ای از اطلاعات تست در دسترس قرار دارد؛ ولی در تست دستی، ایجاد مستندسازی برای تست زمان‌گیر است و عملاً در دستور تست‌کنندگان قرار نمی‌گیرد و انجام نمی‌شود.

با این حال، برخی تست‌ها را نمی‌توان به‌سادگی ماشینی کرد؛ بعضی تست‌ها قابلیت تکرارپذیری ندارند و نمی‌توان آنها را روال‌مند کرد. ابزار ماشینی‌کردن همیشه در اختیار نیست یا مستلزم هزینه هستند. همچنین، گاهی برای نوشتن برخی تست‌ها، به کارشناسان برنامه‌نویسی نیاز

یکی از روش‌های جعبه سیاه، تست تقسیم‌بندی به گروه‌های هم‌ارز است. در این روش، دامنه داده‌های ورودی برنامه به دسته‌های متفاوتی تقسیم می‌شوند و حالات تست، بر اساس این دسته‌ها انتخاب می‌شوند. هدف اصلی استفاده از این روش، کم‌کردن تعداد حالات تست است

است تا مورد تست را ماشینی کنند.

## UI چیست؟

UI، مخفف کلمه User Interface می‌باشد که به «رابط کاربری» معنا می‌شود؛ بر خلاف UX که بر احساس کلی کاربر از یک محصول معطوف است. UI به چگونگی صفحه‌بندی محصول توجه دارد. آنها مسئول طراحی هر صفحه با ابزارها و صورتی هستند که بتواند مسیری که طراح UX طراحی نموده را پیاده‌سازی و جای‌گذاری نماید؛ به عنوان مثال، زمانی که یک طراح UI می‌خواهد یک داشبورد سیستم آمارگیری را پیاده‌سازی کند، ممکن است تصمیم بگیرد که مهم‌ترین موضوع صفحه را در بالای صفحه قرار دهد و یا آن را در نوار کناری صفحه و یا با طراحی یک دکمه آن را به گونه‌ای که به تجربه مشاهده‌ای کاربر و کلیت طرح کمک نماید، نمایش بدهد.

همچنین، طراحان UI معمولاً وظیفه یکپارچگی سبک طراحی در کل روند محصول را نیز بر عهده دارند؛ به صورتی که به صورت شهودی، نوع چینش و محتوای صفحات با یکدیگر همخوانی داشته باشد و همچنین، زبان طراحی مورد استفاده سازگاری خودش را در تمامی روندها و صفحات محصول حفظ نماید. حفظ این ثبات در نمایش المان‌ها، نمایش خطاها و سبک طراحی ضروری می‌باشد.

رابط کاربری نرم‌افزارها، نقطه ارتباط رایانه با کاربران به شمار می‌رود. به واسطه رابط کاربر است که تعامل بین کاربر و نرم‌افزارها شکل می‌گیرد (خالقی، ۱۳۸۵؛ مجد، ۱۳۸۳). در نتیجه، به هر میزان، کاربران بتوانند تعامل آسان، سریع و قابل فهمی با نرم‌افزارها داشته باشند، کارایی نرم‌افزار بالاتر می‌رود و رضایت‌مندی کاربران از آن در سطح بالاتری قرار می‌گیرد. خالقی (۱۳۸۵) در مقاله خود، ضمن تأکید بر تأثیر رابط کاربر در جذب کاربران و مطلوبیت نرم‌افزارها، برای آنها معیارهایی عمومی را برمی‌شمرد که رعایت آنها در رابط کاربری تمامی نرم‌افزارها - یا شاید تمامی محیط‌های رایانه‌ای که قرار است با انسان تعامل داشته باشند - می‌تواند به بهبود عملکرد نرم‌افزار و دستیابی به هدفش در تعامل با کاربر کمک کند. این معیارهای عمومی عبارت‌اند از:

- ثبات: هر چه صفحات و علمکردهای نرم‌افزار از ثبات بیشتری برخوردار باشند، زمان کمتری برای یادگیری و انس کاربر با محیط نرم‌افزار نیاز است. نمونه ثبات در طراحی رابط کاربری در محصولات شرکت مایکروسافت، در مجموعه نرم‌افزارهای آفیس وجود دارد.

- **سادگی و وضوح:** سادگی رابط کاربر، علاوه بر آنکه به یادگیری و استفاده کمک می‌کند، یاریگر ثبات طراحی نیز هست. با دوری از طراحی پیچیده در رابط کاربر نرم‌افزارها، می‌توان وضوح و شفافیت را در مسیرهایی که کاربر باید برای انجام کارکرد مورد نظرش پی‌گیر شود، افزایش داد.

- **رهنمونی:** ویژگی رهنمونی، به هشدارها و جملاتی اشاره دارد که وظیفه راهنمایی کاربر را در هنگام عملکردهای گوناگون به عهده دارد. پیشنهادهای جست‌وجویی که گوگل بعد از جست‌وجو با املاهای غلط به ما ارائه می‌دهد، از این دسته‌اند.

- **زیبایی:** بدون شک، چشم‌نواز بودن رابط کاربری در عین سادگی و آسانی عملکرد، می‌تواند راهبرد مؤثری جهت جذب و تأمین رضایت کاربران باشد.

- **گرافیک و ترکیب رنگ‌ها:** لحاظ کردن مفهوم رنگ‌ها با در نظر گرفتن تفاوت‌های فرهنگی، در معناداری و تناسب رابط کاربری تأثیر به‌سزایی دارد.

- **استفاده از پیام‌ها و علائم اخطاردهنده:** هشدارها و اخطارهای به‌موقع و به‌جا، از ارکان مهم طراحی رابط کاربر نرم‌افزارهاست که عدم توجه به آن، می‌تواند نارضایتی کاربران را به همراه داشته باشد.

- **انعطاف‌پذیری:** برطرف کردن نیازمندی‌های متفاوت کاربرانی با سلیقه‌های متفاوت را می‌توان به عنوان یکی از معیارهای مهم رابط کاربری‌های قوی برشمرد. تنوع کارکردها و سازگاری نرم‌افزارها با برنامه‌های دیگر، از جمله اساسی‌ترین ملاک‌های انتخاب کاربران از بین نرم‌افزارهای مختلف در یک حوزه است.

- **توجه به اصول روان‌شناسی:** آسانی و راحتی کاربری نرم‌افزارها، از جمله تبعات توجه به اصول روان‌شناسی است و ملاک مهمی برای موفقیت نرم‌افزارها به شمار می‌رود.

- **سایر معیارها:** تسلط به محیط، صراحت، کاهش حجم کار کاربران و مساعدت، از دیگر ملاک‌هایی است که خالقی (۱۳۸۵) به عنوان معیارهای عمومی طراحی رابط کاربر برمی‌شمرد.

## UX چیست؟

UX، مخفف کلمه User Experience می‌باشد که معنای «تجربه کاربر» می‌باشد. در حقیقت، طراحان «تجربه کاربر» یا همان UX Designers تمام تلاش خود را می‌کنند تا بهترین سناریو برای آشنایی یک کاربر با یک



محصول و یا خدمات را ایجاد نمایند. از آنجا که این موضوع فرمول خاصی ندارد و متناسب با سوژه‌های مورد نظر ممکن است راهکارهای فراوانی داشته باشد، کار زمان‌بر و دشواری است. یک طراح «تجربه کاربر» می‌خواهد مشکلات کاربر را در مواجهه با سوژه مورد نظر خود، چه از نظر کلامی و لغتی و چه از نظر محل قرارگیری موارد و زیبایی‌شناختی، مورد بررسی قرار دهد و در پی حل آنها برآید تا به بهترین تجربه کاری کاربر یا همان Best User Experience دست یابد.

### مدیریت کیفیت پروژه

مدیریت کیفیت، یکی از بحث‌های جدید در زمینه تولید و خدمات است. گرچه سابقه آن به ابتدای تاریخ نیز می‌رسد، اما مفهوم نوین کیفیت، محصول قرن بیستم است. بر طبق یکی از نگرش‌ها، مدیریت کیفیت، یعنی نظارت بر فرآیند ساخت و تولید محصول برای ایجاد اطمینان از مطابقت محصول با آنچه مورد نظر طراح یا مشتری بوده است. این نظارت، از مرحله دریافت و سفارش مواد اولیه تا خدمات پس از فروش را شامل می‌شود. بدین ترتیب، طیف وسیعی از فعالیت‌ها را شامل می‌گردد. (۱) از جمله فعالیت‌های مربوط به مدیریت کیفیت، یکی، تضمین کیفیت و دیگری، کنترل کیفیت است. امروزه، مدیریت کیفیت، نه تنها شامل تولید، بلکه شامل خدمات و حتی خدمات دولتی نیز می‌شود. مدیریت کیفیت دولت‌ها و آموزش و پرورش (Graduation)، یکی از مباحث نوین در کیفیت است. مدیریت کیفیت، عموماً شامل برقراری خط مشی کیفیت و اهداف کیفیت، طرح‌ریزی کیفیت، کنترل کیفیت، تضمین کیفیت و بهبود کیفیت است.

مدیریت کیفیت پروژه، دربرگیرنده فرایندهای مورد نیاز برای حصول اطمینان از برآورده شدن نیازهایی است که پروژه برای آن تعهد شده است. مدیریت کیفیت پروژه، «کلیه فعالیت‌های کارکرد مدیریت عمومی را که خط مشی و مسئولیت کیفیت را تعیین نموده و آنها را به‌وسیله ابزارهایی چون: برنامه‌ریزی کیفیت، تضمین کیفیت، کنترل کیفیت و بهبود کیفیت در قالب سیستم کیفیت اجرا می‌نماید» شامل می‌شود. مدیریت کیفیت در pmbok زیرمجموعه‌ای از مدیریت پروژه است.

– برنامه‌ریزی کیفیت: یعنی شناسایی استانداردهای کیفیتی مرتبط با پروژه و تعیین چگونگی تحقق آنها.  
– تضمین کیفیت: یعنی ارزیابی عملکرد کلی پروژه بر

اساس یک مبنای منظم، به منظور حصول اطمینان از اینکه پروژه استانداردهای کیفیت مرتبط را محقق خواهد ساخت.  
– کنترل کیفیت: یعنی نظارت بر نتایج مشخص پروژه، به منظور تعیین انطباق آنها با استانداردهای کیفیت مرتبط و شناسایی راه‌هایی برای حذف علل عملکرد ناخوشایند. این فرایندها با یکدیگر و همچنین با فرایندهای سایر حوزه‌های دانش تعامل دارند. ممکن است، هر فرایند بر مبنای نیازهای پروژه تلاش یک یا تعداد بیشتری از افراد یا گروه‌هایی از آنان را دربرداشته باشد. هر فرایند به طور معمول، دست کم یکبار در هر مرحله پروژه، به وقوع می‌پیوندد.

ویژگی‌های کیفیت نرم‌افزار (۲) از دیدگاه کاربر عبارت‌اند از:

۱. میزان پوشش‌دهی وظایف تعیین‌شده در مستندات نیازمندی‌ها و موارد ضمنی (Functionality)؛
۲. قابلیت اطمینان یا قابل اطمینان بودن (Reliability)؛
۳. قابلیت استفاده یا قابل استفاده بودن (Usability)؛
۴. کارایی یا به عبارتی دیگر، پربازده و مؤثر بودن (Efficiency)؛
۵. قابلیت نگهداشت (Maintenance)؛
۶. قابلیت حمل یا به عبارتی دیگر، قابل انتقال بودن (Portability).

### ارزیابی نرم‌افزار

ارزیابی، فرایندی است که به جمع‌آوری اطلاعات درباره ارزش یک برنامه با هدف تصمیم‌گیری درباره اثربخشی یا بهبود آن برنامه صورت می‌گیرد (استون، ۲۰۰۸)؛ به بیانی دیگر، ارزیابی، «اندازه‌گیری عملکرد یا خدمات یک نظام و بررسی اثر آن در پیشبرد اهداف تعیین‌شده» است (سلطانی و راستین، ۱۳۷۹). ارزیابی می‌تواند به عنوان وظیفه‌ای در نظر گرفته شود که به یک یا چند خروجی می‌انجامد و در عین حال، هدفی برای برنامه‌ریزی به شمار می‌آید. افزون بر این، ویژگی دیگر ارزیابی، «هدف‌محور» بودن آن است. ارزیابی در زمینه گسترش نرم‌افزارها، از دهه‌های اخیر که نقش ارزیابی در تعامل انسان و رایانه تغییر یافته، به عنوان یک هدف کاربرد داشته است (گدیگا، هامبورگ و دونش، ۲۰۰۲ به نقل از ساچمن، ۱۹۶۷). تولید روزافزون محصولات نرم‌افزاری با هدف برطرف کردن نیازهای جدید کاربران، و در نتیجه، نیاز به ارتقا و تکمیل محصولات نرم‌افزاری، اهمیت ارزیابی نرم‌افزارها را دوچندان نموده است. نرم‌افزارها را می‌توان از دیدگاه‌های متفاوتی مانند: کارکرد، قابلیت اطمینان، کاربردپذیری، بازدهی، قابلیت

برطرف کردن نیازمندی‌های متفاوت کاربرانی با سلیقه‌های متفاوت را می‌توان به عنوان یکی از معیارهای مهم رابط کاربری‌های قوی برشمرد. تنوع کارکردها و سازگاری نرم‌افزارها با برنامه‌های دیگر، از جمله اساسی‌ترین ملاک‌های انتخاب کاربران از بین نرم‌افزارهای مختلف در یک حوزه است

نگه‌داری و یا حمل‌پذیری بررسی کرد. فرایند ارزیابی نرم‌افزار، در عمل می‌تواند دست‌کم یکی از این سه هدف را دنبال کند: مقایسه نرم‌افزارها، تعیین میزان کیفیت و یا تعیین نقاط ضعف آنها (گدیگا و همکاران، ۲۰۰۲). (۳)

ارزیابی نرم‌افزار، عبارت است از فرایند بررسی یا اجرای یک نرم‌افزار یا جزء نرم‌افزاری به صورت دستی یا خودکار، به منظور ارزیابی تطابق ویژگی‌های آن با نیازهای از قبل مشخص شده و یا به منظور مقایسه بین نتایج مورد انتظار و نتایج واقعی. (۴)

### انواع ارزیابی نرم‌افزار

نرم‌افزارها را از دیدگاه‌های گوناگون می‌توان مورد ارزیابی قرار داد. نوع، کارکرد و هدف خود نرم‌افزار هم می‌تواند تعیین‌کننده نوع ارزیابی باشد. به اعتقاد عده‌ای از صاحب‌نظران، ارزیابی نرم‌افزار، فرایندی هدف‌محور است (کومی - سیرویو، ۲۰۰۴؛ کیچنهام، لینکمن و لائو، ۱۹۹۶؛ گدیگا و همکاران، ۲۰۰۲). مراحل ارزیابی، زمان، هزینه و تمامی فرایند ارزیابی، بستگی به هدفی دارد که به واسطه آن انجام ارزیابی صورت می‌گیرد. ارزیابی نرم‌افزارها می‌تواند با اهداف متفاوتی صورت پذیرد؛ برای مثال، نرم‌افزاری را در نظر بگیرید که با هدف نیازسنجی کاربران برای گنجاندن یک سرویس اضافی در سرویس‌های خود، بررسی عملی این امکان را به همراه ارزیابی نظرات کاربران در این مورد آغاز می‌کند. در مقابل، نرم‌افزاری برای ارزیابی نظرات کاربران درباره دسته‌ای از خدمات خود به کار گرفته می‌شود تا در صورت نیاز، ارتقای خدمات خود را در تولید نرم‌افزار لحاظ نماید. مطمئناً اهداف ارزیابی در این دو نرم‌افزار، یکسان نیست و مراحل، زمان و هزینه‌های متفاوتی خواهند داشت. در مورد هدف ارزیابی نرم‌افزارها، حتی می‌توان مواردی را

بیان کرد که ارزیابی نرم‌افزار به منظور مقایسه و انتخاب آنها برای انطباق با محیط و عملکردهای خاصی که مد نظر یک فرد یا یک سازمان است، انجام می‌گیرد. با وجود گزینه‌های مختلف نرم‌افزارهایی که برای رفع نیازهای فرد یا سازمان وجود دارد، ارزیابی نرم‌افزارها پیش از انتخاب و خرید و بررسی مطابقت آنها با نیازهای سازمان، اهمیت قابل توجهی دارد و موجب صرفه‌جویی در زمان و هزینه‌های فردی و سازمانی می‌شود. رولی (۱۹۹۲) معتقد است، باید به چنین فرایندهایی که با هدف بررسی و مقایسه نرم‌افزارها به منظور انتخاب و خرید آنها انجام می‌شود، از منظر یک پروژه نگاه کرد. برای این کار، باید بعد از مشخص ساختن اهداف، روش‌شناسی مناسب را اتخاذ نمود و در مرحله بعد، معیارهای متناسب را استخراج کرد و فرایند بررسی را انجام داد. در زمینه مقایسه نرم‌افزارهای جایگزین، فر، مودیگوندا و الامی (۲۰۰۹) نیز روشی آماری برای مقایسه ارائه داده‌اند. از لحاظ استمرار فعالیت‌های ارزیابی نرم‌افزار، به طور کلی، می‌توان دو دسته ارزیابی را تشخیص داد: ۱. ارزیابی‌های مستمر؛ ۲. ارزیابی مقطعی.

میوشی، تاگوشی و آزوما (۱۹۸۹)، در یکی از متون قدیمی مربوط به ارزیابی نرم‌افزارها اشاره می‌کنند که معمولاً ارزیابی نرم‌افزار، توسط تولیدکنندگان آن نرم‌افزارها صورت نمی‌گیرد. آنها بر پایه اطلاعات استخراج‌شده از پروژه عملی فست (FASET)، مدلی را ارائه می‌دهند که به صورتی روشمند، ارزیابی محیط‌های گسترش نرم‌افزار را انجام می‌دهد و به تولیدکنندگان نرم‌افزارها کمک می‌کند تا محصولات خود را مورد ارزیابی قرار دهند. در مدل آنها، مراحل مشخصی باید طی شود تا ارزیابی نرم‌افزارها از دیدگاه کیفی امکان‌پذیر شود:

\* مشخص ساختن اهداف ارزیابی؛

\* انتخاب ویژگی‌های کیفی، مطابق با چرخه عمر اهداف ارزیابی؛

\* تصحیح و پالایش تمام ویژگی‌های کیفی؛

\* تعریف ساختار کیفی؛

\* انتخاب و تعریف سنجها؛

\* آزمون اهمیت یا وزن هر کدام از ویژگی‌های کیفی؛

\* آزمون الگوریتم امتیازدهی؛

\* تعریف رهنمودهایی برای بررسی؛

\* آزمون چگونگی ارائه نتایج ارزیابی.

### تحلیلگر کیفیت نرم‌افزار

تحلیلگر کیفیت نرم‌افزار، مسئول اعمال اصول و شیوه‌های تضمین کیفیت نرم‌افزار در سراسر چرخه عمر توسعه نرم‌افزار است. نقش تحلیلگر کیفیت نرم‌افزار، بیشتر با نقش نرم‌افزار تست اشتباه گرفته می‌شود. معمول شرکت‌های نرم‌افزاری، تعیین تست نرم‌افزار را به عنوان تضمین‌کننده کیفیت نرم‌افزار می‌بینند؛ درحالی‌که این نقش‌ها متفاوت هستند.

نرم‌افزار تست محصول‌گرا، تضمین کیفیت نرم‌افزار فرایندگراست. نقش در تضمین کیفیت نرم‌افزار، به عنوان مهندسی نرم‌افزار تضمین کیفیت/کیفیت نرم‌افزار تحلیلگر/مشاور فرایند، تعیین شده است. برخی از وظایف تحلیلگران کیفیت نرم‌افزار، عبارت‌اند از:

شرکت در جلسات بررسی مدیریت کیفیت؛ تأیید اسناد QMS برای مدیر QA که نویسنده اصلی است؛ انجام ممیزی داخلی کیفیت؛ برنامه و انجام ممیزی‌های کیفیت از پیمانکاران فرعی؛ حفظ و به‌روزرسانی پایگاه داده آموزش و ممیزی به ارائه گزارش‌ها؛ شناسایی مشکلات و یا نقص در محصولات و سیستم مدیریت کیفیت؛ حل مشکلات مسائل پیاده‌سازی QMS.

### فرصت‌ها و ظرفیت‌های کنترل کیفیت

موضوع کیفیت و بهبود آن، موضوع مهم و حیاتی در هر سازمان و شرکت است. برخی کیفیت را تنها انطباق با ضوابط از پیش تعیین شده می‌دانند؛ به عبارت دیگر، کیفیت، مترادف با استاندارد تلقی می‌شود؛ غافل از آنکه امروزه کیفیت، مفهومی وسیع‌تر داشته و به معنای پاسخگویی به درخواست‌های منطقی و حق‌مدارانه مشتریان نیز هست. اگرچه انطباق با استاندارد نیز در سطوح ملی و بین‌المللی

می‌تواند درجه‌ای از کیفیت محسوب شود، اما همه ابعاد کیفیت نیست. بر این اساس، در شرایط کنونی، هدف از کیفیت علاوه بر اجرای استانداردها، کسب رضایت حداکثری مشتری در همه ابعاد محصول را نیز شامل می‌شود.

شرکت‌ها و مؤسساتی که در تولید نرم‌افزار در سطح جهان فعالیت می‌کنند، شیوه‌ها و روش‌های مختلفی برای خود دارند که در چرخه طول عمر نرم‌افزار از آنها استفاده می‌کنند تا نرم‌افزاری طول عمرش تمام بشود و بایگانی یا نسخه تازه‌تری از آن ارائه شود. پیش‌تر نیز از این موضوع صحبت کردیم که بیشتر شرکت‌های بزرگ نرم‌افزاری یا در حوزه فناوری، اداره یا بخشی به نام کنترل کیفیت محصولات را دارا هستند؛ اما با توجه به نوع محصولاتشان، کارکردهای متفاوتی دارند. در واقع، کنترل کیفیت این فرصت را ایجاد می‌کند تا کارشناسانی که فناور، آگاه و مسلط به محصول هستند و از نیازهای کاربر نیز آگاهی داشته و بر مراحل تولید نرم‌افزار هم نظارت دارند، بتوانند به‌خوبی محصول را مورد سنجش قرار دهند و در هزینه و وقت صرفه‌جویی کنند.

طبق نظر پروفسور جوران، «کیفیت» دارای دو مفهوم مکمل است:

اول: هدف، کیفیت بالاتر، ایجاد رضایت بیشتر در مشتری و انتظار درآمد بالاتر است. ایجاد ویژگی‌های بیشتر و بهتر کیفیت، مستلزم سرمایه‌گذاری و افزایش هزینه است. از این زاویه، کیفیت بالاتر، هزینه‌بردار است؛ اما برای درآمد افزون‌تر.

دوم: کیفیت، یعنی عدم عیب و خطا، عیب و خطا موجب ضایعات، تعمیرات، دوباره‌کاری، محصول مرجوع، نارضایتی مشتری، شکایت، جریمه و از دست‌دادن مشتری می‌گردد و همه اینها، هزینه‌آور هستند. بدین ترتیب، کیفیت بالاتر، حاوی عیب و خطای کمتر و در نتیجه، هزینه کمتر است.

کنترل کیفیت نرم‌افزار که در استانداردهای مختلف با عناوین گوناگون از آن تعبیر می‌شود، باید بر هر محصولی که می‌خواهد در چرخه تولید توسعه و پشتیبانی قرار گیرد، نظارت داشته باشد؛ یعنی اول مشخص شود که این محصول برای تولید و توسعه پشتیبانی و بایگانی خود، از چه استانداردی استفاده می‌کند و بعد از آن باید بر این موضوع که محصول بر طبق استاندارد مورد نظر تولید و توسعه پیدا می‌کند، نظارت داشته باشد که تمامی آیتم‌های آن استاندارد رعایت شود و اگر بخواهند از این آیتم‌ها عدول کنند، کنترل کیفیت باید در جریان آن باشد و دلایل آن مشخص شود.



و تصمیم‌گیری درباره نوع تستی است که باید انجام شود؛ برای مثال، چک‌لیست تست دستی و خودکار تهیه شود و اینکه هر کدام در چه زمانی انجام شود، بهتر است و گزارش مناسبی به‌دست خواهد آمد.

### ۳. طراحی تست (test design):

طرح‌ها و موارد تست باید بازبینی شوند و ماتریس ارزیابی عملکرد نیز بازبینی و نهایی گردد. در این مرحله، معیار ارزیابی تست نیز انجام می‌شود و موارد تستی را که باید اتومات شود، اتومات کنیم. داده‌های تست نیز باید جمع‌آوری شود. برنامه‌های تست، در این مرحله مشخص می‌شوند.

### ۴. ساخت و ارزیابی (-construction and verification):

در این فاز، باید همه طرح‌ها و موارد تست کامل شود و باید گزارش باگ‌هایی که توسط بازبین‌ها و کارشناسان به‌دست می‌آید، به دو قسمت نتایج تست‌های خودکار و دستی مشخص باشد. تست‌ها همه در این فاز انجام می‌شوند و گزارش‌ها به‌دست می‌آیند.

### ۵. چرخه‌های تست (testing cycles):

در این فاز تمام فازهای قبل توسط کارشناس مورد بازبینی قرار می‌گیرد تا مطمئن شود تمام مواردی که تعیین شده یک به یک انجام شده است یا خیر؛ چک‌لیست تست‌ها، گزارش باگ‌ها، موارد و معیارهای تست، افزودن تست‌های جدید (در صورت نیاز)، بازبینی مجدد تست‌ها برای اطمینان از صحت.

### ۶. تست‌کردن نهایی و پیاده‌سازی (The final test):

در این مرحله، باید تست استرس و کارایی باقیمانده را اجرا کنیم و مستندات تست باید تکمیل شوند. همچنین، تست

در مدیریت کیفیت و مهندسی و تولید صنعتی، بخش کنترل کیفیت (Quality Control، به اختصار QC)، و مهندسی کیفیت به بخشی گفته می‌شود که به درست کردن روش‌هایی مشغول است تا کارخانه بتواند به‌وسیله آن روش‌ها، از مرغوبیت و مشتری‌پسند بودن کالاهای تولیدی خود مطمئن گردد. این روش‌ها و سیستم‌ها، معمولاً با همکاری با دیگر رشته‌های مهندسی و بازرگانی طراحی می‌شوند. کنترل کیفیت، مجموعه عملیاتی نظیر اندازه‌گیری یا آزمون است که روی یک محصول یا کالا انجام می‌شود تا مشخص شود آیا آن محصول با مشخصات فنی مورد نظر مطابقت دارد یا خیر. گروه کنترل کیفیت، گروه کوچکی از کارکنان است که ماهیت کار مشابه دارند و برای حل مشکلات و یا بهبود فرایند کاری در راستای اهداف و فعالیت‌های کنترل کیفیت سازمان، به صورت داوطلب و با انگیزه درونی و بدون هیچ‌گونه نظارت خارجی و از طریق تشکیل جلسات منظم و حضوری در اوقات فراغت برای افزایش بازده کیفی شرکت یا کارخانه خود تلاش می‌کنند. (۵)

### چرخه آزمون نرم‌افزار

چرخه تست نرم‌افزار، از هفت فاز کلی تشکیل شده است:

#### ۱. طرح‌ریزی تست (test planning):

این فازی است که مدیر کیفیت و مدیر پروژه باید تصمیم بگیرند که چه مواردی باید تست شوند. طبیعتاً طرح‌ریزی کافی در این مرحله، به میزان زیاد ریسک نرم‌افزارهای بدون کیفیت را کاهش می‌دهد.

#### ۲. تحلیل تست (test analysis):

زمانی که طرح‌ریزی تست تحقق یافت و روی آن تصمیم‌گیری شد، مرحله بعدی، کمی کاوش پیرامون پروژه

ابزارهایی که فرآیند تولید نرم‌افزار را بهینه می‌کنند، با استانداردهای خاص خود درباره چگونگی تولید نرم‌افزار و عملیاتی است که درون PLC باید اتفاق بیفتد تا به حد مطلوب برسد. روش‌های مختلف ارزیابی، برای آن است تا چیدمان غیربهینه و تخصص و ابزار نامناسب به‌کار برده نشود

بار نیز انجام می‌شود و برنامه نیز پس از یکبار تصحیح، مجدداً مورد بازبینی قرار می‌گیرد.

**۷. تست پس از پیاده‌سازی (post implementation):**  
در این فاز، فرایند تست ارزیابی می‌شود و مستندات و تجربیات ثبت می‌شوند. تمام ارزیابی‌ها و مسائل مشابه برای استفاده در پروژه‌های آینده، مشخص و شناسایی و ثبت می‌شوند.

### برنامه‌ریزی، تضمین و کنترل کیفیت

برنامه‌ریزی کیفیت، متضمن شناسایی استانداردهای کیفیت مرتبط با پروژه و تعیین چگونگی تحقق آنهاست. برنامه‌ریزی کیفیت، یکی از فرایندهای کمکی (جانبی) کلیدی در برنامه‌ریزی پروژه می‌باشد.

کنترل کیفیت، متضمن نظارت بر نتایج مشخص پروژه به منظور تعیین مطابقت آنها با استانداردهای کیفیت مرتبط و شناسایی راه‌هایی برای حذف عوامل نتایج ناخوشایند می‌باشد. این فرایند باید در طول پروژه اجرا گردد. کنترل کیفیت، بیشتر توسط بخش کنترل کیفیت یا با نامی مشابه اجرا می‌گردد.

تضمین کیفیت، عبارت است از کلیه فعالیت‌های برنامه‌ریزی شده و نظام‌یافته پیاده‌شده در چارچوب سیستم کیفیت به منظور حصول اطمینان از اینکه پروژه استانداردهای کیفیت مربوطه را محقق خواهد ساخت.

### ابزارها و فنون کنترل کیفیت

**۱. بازرسی:** بازرسی، مشتمل است بر فعالیت‌هایی از قبیل: اندازه‌گیری، آزمایش و آزمون که برای تعیین مطابقت نتایج با الزامات، تعهد شده‌اند. بازرسی‌ها (بررسی) ممکن است، در هر سطحی انجام شوند. بازرسی‌ها به اشکال مختلفی چون: بازرسی‌ها، ممیزی‌ها و بررسی‌ها نامیده شده‌اند.

**۲. نمودارهای کنترلی:** منظور، نمایش تصویری از نتایج یک فرایند در بستر زمان می‌باشند. از این نمودارها جهت تعیین تحت کنترل بودن فرایند در بستر زمان استفاده می‌گردد. انحرافات و رویدادهای غیرمعمول، در این نمودار آشکار می‌شود.

**۳. نمونه‌گیری آماری:** نمونه‌گیری آماری، متضمن انتخاب بخشی از جامعه مورد توجه جهت بازرسی می‌باشد.

**۴. جریان نمایی:** جریان نمایی در کنترل کیفیت، به منظور کمک به تحلیل چگونگی وقوع مشکلات به کار گرفته

می‌شود.

**۵. تحلیل روند:** دو موضوع را تحت نظر می‌گیرد:

- عملکرد فنی: اینکه چه تعداد خطا شناسایی شده‌اند و چه تعداد اصلاح‌نشده باقی مانده‌اند.

- عملکرد هزینه و زمان: در هر دوره زمانی، چه تعداد از فعالیت‌ها با مغایرت‌های قابل توجهی خاتمه یافته‌اند.

### خروجی‌های کنترل کیفیت

این خروجی‌ها عبارت‌اند از:

\* بهبود کیفیت؛

\* تصمیمات‌پذیرشی؛

\* دوباره کاری؛

\* فهرست‌های واری؛

\* تنظیمات فرایند؛

\* مستندات.

### پی‌نوشت‌ها:

۱. این نگرش، مطابق استاندارد ISO 9001 می‌باشد.

۲. هاشمی، ۱۳۸۹. «استاندارد مهندسی نرم‌افزار؛ رهنمود عمومی در تعریف کیفیت محصولات نرم‌افزاری». شورای عالی انفورماتیک.

۳. رجب‌زاده، امیرحسین، ۱۳۹۲. «ارزیابی نرم‌افزار؛ انواع و کارکردها». فلصنامه ره‌آورد نور، ش ۴۲: ۳۹ - ۳۹.

۴. روحانی، مرجان، ۱۳۹۲. «مبانی و مفاهیم نظارت و ارزشیابی سنجش و عملکرد کارایی و اثربخشی برنامه‌ها». شورای عالی انفورماتیک، پاییزماه.

۵. فقیه، نظام‌الدین، مهندسی کیفیت و اطمینان.

### منابع:

۱. پایگاه اینترنتی ویکی‌پدیا، ذیل عنوان «تست نرم‌افزار»:

[www.wikipedia.org](http://www.wikipedia.org)

۲. —. ذیل عنوان «تحلیلگر کیفیت نرم‌افزار»:

[www.wikipedia.org](http://www.wikipedia.org)

۳. اصولی، سید حسین و احسان نجابت، علی بیاتی، حسین ناصری و علی افخمی. **پیکره دانش مدیریت پروژه**. انجمن PMI. ■