



۱- کدام نوع از انواع طراحی نرم افزار بیانگر نحوه ارتباط کاربر نهایی با سیستم است؟

- (۱) طراحی کلاس/داده
- (۲) طراحی معماری
- (۳) طراحی واسط
- (۴) طراحی در سطح مولفه (Component)

۲- راهبرد تقسیم و حل (Divide and Conquer) بر این حقیقت استوار است که شکست یک مساله ی بزرگ به مسائل کوچک تر و حل آن مسائل کوچک تر، از حل کردن مساله ی بزرگ آسان تر خواهد بود. در کدام یک از مفاهیم بنیانی در طراحی، این راهبرد دیده نمی شود؟

- (۱) انتزاع
- (۲) تفکیک دغدغه ها (Separation of Concerns)
- (۳) ماژولار بودن
- (۴) استقلال عملکردی

۳- از نمودار فعالیت (Activity Diagram) برای مدلسازی کدام یک از انواع المان های طراحی استفاده می شود؟

- (۱) المان های واسط
- (۲) المان های مولفه ای
- (۳) المان های طراحی داده
- (۴) المان های معماری

۴- کدام مورد درباره ی Refactoring درست نیست؟

- (۱) در عین بهبود ساختار داخلی کد رفتار بیرونی آن را تغییر نمی دهد.
- (۲) در حین آن، طراحی موجود از اضافه کاری، المان های طراحی اضافی پاک سازی می شود.
- (۳) نتیجه ی حاصل از refactoring، کدی است که تجمیع، تست و نگهداری آن ساده تر است.
- (۴) ممکن است در حین refactoring، الگوریتم های پیچیده برای کاهش حجم کد افزوده شوند.

۵- ارائه‌ی یک سرویس، از طریق تنها یک متد برای کلاس طراحی، به کدام خصوصیت کلاس‌های طراحی well-formed اشاره دارد؟

- (۱) کامل و کافی بودن
- (۲) بدوی بودن (Primitiveness)
- (۳) انسجام بالا
- (۴) جفت‌شدگی پایین

۶- کدام گزینه در مورد Modularize کردن در طراحی درست است؟

- (۱) افزایش ماژول‌ها در طراحی باعث بالا رفتن هزینه‌ی جمع‌بندی می‌شود.
- (۲) محاسبه‌ی تعداد بهینه ماژول‌ها در طراحی به سادگی و بدون دانش راجع به دیگر جنبه‌های طراحی قابل انجام است.
- (۳) با افزایش Modularity در طراحی، اعمال تغییرات در نرم‌افزار سخت‌تر می‌شود.
- (۴) آزمون‌پذیری کم، همزمان با اعمال Modularity کاهش می‌یابد.

۷- یکی از بهترین نتایجی که از این مفهوم طراحی به دست می‌آید این است که "اعمال تغییرات در هنگام تست و نگهداری از سیستم بسیار ساده‌تر می‌شود. زیرا خطاهایی که در اثر تغییرات به وجود می‌آیند، به مکان‌های دیگر سیستم انتشار نمی‌یابند". این مفهوم کدام یک از مفاهیم طراحی زیر است؟

- (۱) معماری
- (۲) الگوها
- (۳) تفکیک دغدغه‌ها
- (۴) پنهان کردن اطلاعات

۸- کدام مورد از دلایل ایجاد معماری نرم‌افزار نیست؟

- (۱) شناخت جزئیات نرم‌افزار
- (۲) بررسی کارایی طراحی
- (۳) بررسی طراحی‌های جایگزین
- (۴) کاهش ریسک ایجاد نرم‌افزار

۹- علی آقا که یک مهندس نرم‌افزار زبردست است، به تازگی به پروژه‌ی اصلی سازمان ما اضافه شده است. به نظر شما اولین کاری که برای آشنایی او با پروژه باید انجام دهیم، کدام است؟

- (۱) مخزن کد را در اختیار او بگذاریم تا آن را بخواند و با پروژه آشنا شود.
- (۲) اعضای تیم توسعه (اعم از کسانی که اکنون سر کار هستند و کسانی که پیش‌تر در تیم بودند) را به او معرفی کنیم تا با صحبت با آن‌ها، با پروژه آشنا شود.
- (۳) مستندات معماری سطح بالای نرم‌افزار را به او نشان دهیم تا با پروژه آشنا شود.
- (۴) نیازی به آشنایی اولیه نیست. در طول توسعه‌ی نرم‌افزار به تدریج با پروژه آشنا می‌شود.

۱۰- هر سبک (Style) معماری شامل چه چیزی نیست؟

- ۱) مولفه‌ها (Components)
- ۲) ارتباطات (Connectors)
- ۳) جریان داده (Dataflow)
- ۴) محدودیت‌ها (Constraints)

۱۱- معماری سه‌لایه از کدام لایه تشکیل نشده است؟

- ۱) نمایش (Presentation)
- ۲) منطق کسب‌وکار (Business logic)
- ۳) زیرساخت (Infrastructure)
- ۴) دسترسی به داده (Data access)

۱۲- فرض کنید در میانه‌ی راه توسعه‌ی پروژه هستیم. برای اصلاح یک مشکل در نرم‌افزار، تغییر کدام مورد حتماً هزینه‌برتر است؟

- ۱) کلاس‌های طراحی
- ۲) نیازمندی‌های وظیفه‌ای مشتری
- ۳) معماری سطح‌بالای نرم‌افزار
- ۴) طراحی‌های رابط کاربری نرم‌افزار

۱۳- شغل «مشاور املاک» به کدام یک از الگوهای معماری زیر شبیه‌تر است؟

- 1) Client-server pattern
- 2) Microkernel pattern
- 3) Publish-subscribe pattern
- 4) Broker pattern

۱۴- «کلاس فرزند پیش‌شرط‌های بیشتر و پس‌شرط‌های کمتری نسبت به کلاس پدر نداشته باشد» اشاره به کدام یک از اصول طراحی دارد؟

- 1) Dependency Inversion Principle
- 2) The Release Reuse Equivalency Principle
- 3) The Liskov Substitution Principle
- 4) The Common Reuse Principle

۱۵- استفاده از رابط‌ها به تحقق کدام یک از اصول طراحی کمک می‌کند؟

- 1) The Interface Segregation Principle
- 2) The Open-Closed Principle
- 3) The Release Reuse Equivalency Principle
- 4) The Common Closure Principle

۱۶- طراحی اپلیکیشن تلفن همراه تپسی، به کدام صورت است؟

- ۱) Thin web-based
- ۲) Rich
- ۳) بسته به نوع سیستم عامل تلفن همراه می تواند thin یا rich باشد.
- ۴) این اپلیکیشن به هیچ کدام یک از دو روش بالا طراحی نشده است.

۱۷- کدام مورد از موانع استفاده ی مجدد از مولفه ها نیست؟

- ۱) نبود مستندات کافی جهت استفاده از مولفه
- ۲) استفاده سازمان ها از متدولوژی های توسعه نرم افزار که به استفاده ی مجدد از مولفه ها اهمیتی نمی دهند.
- ۳) عدم تطبیق رابط های مولفه با نیاز پروژه
- ۴) جا نیفتادن فرهنگ استفاده ی مجدد از مولفه ها بین توسعه دهندگان

۱۸- کدام یک از گزینه های زیر تعریفی برای کیفیت ارائه می دهد؟

- ۱) کیفیت یعنی آنچه که شما به سرعت متوجه اش می شوید اما قادر به تعریف کردن واضح آن نیستید.
- ۲) کیفیت یعنی فیچرها و عملکردهایی که یک محصول دارد.
- ۳) کیفیت یعنی کاربر تمام انتظاراتی که از محصول داشته است را در محصول نهایی ببیند.
- ۴) همه موارد

۱۹- تعریفی که در حالت کلی برای کیفیت نرم افزار بیان می شود به صورت زیر است:

*An effective process applied in a manner that creates a useful product that provides measurable value for those who produce it and those who use it.*

با توجه به این تعریف کدام گزینه نادرست است؟

- ۱) فرآیند موثر اشاره دارد به هر گونه زیرساختی که منجر به تولید یک نرم افزار کیفیت بالا شود؛ برای مثال اقدامات مدیریت فرآیند، امکاناتی که مهندسی نرم افزار برای توسعه دهنده ها فراهم می کند و فعالیتهای چتری
- ۲) محصول مفید اشاره دارد به فیچرهای مطلوب کاربران نهایی به نحوی که قابل اتکا و عاری از خطا باشند.
- ۳) ارزش افزوده اشاره به مزایایی دارد که نرم افزار با کیفیت برای سازمان (برای مثال کاهش تلاش های لازم برای نگهداری نرم افزار، رفع باگ کمتری ..) و کاربران نهایی بوجود می آورد.
- ۴) هیچ کدام

۲۰- دیوید گاروین ۸ بعد برای کیفیت نام می برد که در گزینه های زیر برخی از آن ها بیان شده است. کدام گزینه تعریف درست آن بعد را شامل نمی شود؟

- ۱) قابلیت اتکاپذیری: این معیار کیفیت به معنای آن است که نرم افزار رفتار درست و بدون خطایی داشته باشد.
- ۲) تطابق پذیری: این معیار کیفیت به معنای آن است که نرم افزار مطابق با نیازمندی های تعریف شده باشد.

۳) دوام‌پذیری: این معیار کیفیت به معنای آن است نرم‌افزار در مرحله نگهداری بدون تولید اثرات جانبی بتواند تغییر کند و رفع باگ شود.

۴) بازدهی: این معیار کیفیت به معنای آن است که نرم‌افزار تمام فیچرهای مورد انتظار را به نحوی ارائه دهد که برای کاربر ارزشمند باشد.

۲۱- کدام گزینه جز هزینه‌های کیفیت نیست؟

- ۱) هزینه‌های پیشگیری همانند برنامه‌ریزی برای کنترل کیفیت
- ۲) هزینه‌های ارزیابی همانند آزمون نرم‌افزار
- ۳) هزینه‌های شکست همانند کار کردن مجدد
- ۴) هزینه‌های نگهداری نرم‌افزار همانند تغییرات

۲۲- کدام گزینه لزوماً از اثرات یک نرم‌افزار کم کیفیت نیست؟

- ۱) نرم‌افزار کم کیفیت به آسانی هک می‌شود و به محض استقرار در معرض حملات امنیتی است.
- ۲) نرم‌افزار کم کیفیت دارای باگ در معماری و پیاده‌سازی آن است.
- ۳) نرم‌افزار کم کیفیت انتظارات کاربر نهایی را برآورده نمی‌کند.
- ۴) نرم‌افزار کم کیفیت منجر به افزایش هزینه‌های سازمان می‌شود.

۲۳- کدام گزینه موارد موثر در رسیدن به نرم‌افزار با کیفیت را بیان می‌کند؟

- ۱) مدیریت پروژه‌ی خوب - کنترل کیفیت - تضمین کیفیت
- ۲) استخراج نیازمندی‌های خوب - کنترل کیفیت - صرف هزینه‌های زیاد
- ۳) استخراج نیازمندی‌های خوب - مدیریت پروژه خوب - کنترل کیفیت
- ۴) مدیریت پروژه خوب - صرف هزینه زیاد - کنترل کیفیت

۲۴- جمله زیر مفهوم کدام یک از ابعاد کیفیت را بیان می‌کند؟

“کاربر اپلیکیشن جدید را دانلود کرده و پس از ۲ دقیقه تجربه آن می‌تواند به صورت مفید آن را به کار گیرد.”

- ۱) کارایی
- ۲) انعطاف‌پذیری
- ۳) استفاده‌پذیری
- ۴) قابلیت اطمینان

۲۵- کدام یک از تصمیمات مدیریتی زیر تاثیر بیشتری در کیفیت نرم‌افزار دارد؟

- ۱) تصمیمات تخمینی
- ۲) تصمیمات مرتبط با ریسک در پروژه

(۳) تصمیمات زمانبندی

(۴) همه موارد

۲۶- کدام یک از گزینه‌های زیر سنگین‌ترین هزینه برای کیفیت نرم‌افزار محسوب می‌شود؟

(۱) هزینه‌های ارزیابی

(۲) هزینه‌های شکست داخلی

(۳) هزینه‌های شکست خارجی

(۴) هزینه‌های پیشگیری

۲۷- جمله زیر مفهوم کدام یک از ابعاد کیفیت را بیان می‌کند؟

“در حالت عادی، سیستم باید هر تراکنشی که کاربر ایجاد می‌کند را با میانگین تاخیر ۲ ثانیه پردازش می‌کند.”

(۱) قابلیت اطمینان

(۲) کارایی

(۳) قابلیت سرویس‌دهی

(۴) کیفیت ویژگی

۲۸- اصلی‌ترین و اولیه‌ترین انگیزه برای انجام بازبینی‌های فنی (Technical Reviews) چیست؟

(۱) ساده‌سازی کد و حذف الگوریتم‌های پیچیده و غیرضروری

(۲) کنترل کردن این نکته که طراحی شیء‌گرا اعمال شده‌است یا خیر

(۳) انجام Refactoring بر روی کد

(۴) پیدا کردن errorها در حین انجام فرآیند و جلوگیری از بروز Defect

۲۹- فرض کنید در یک شرکت برای کارهای فنی پیچیده جلسات بازبینی با ویژگی‌های زیر تشکیل می‌شود. کدام یک از این ویژگی‌ها

برای جلسات بازبینی مناسب نیست؟

(۱) قبل از جلسه هر کدام از شرکت‌کننده‌ها یک روز کاری کامل را صرف آمادگی برای جلسه می‌کنند.

(۲) در هر جلسه تمام اعضای فنی در همه سطوح شرکت می‌کنند که تعداد آن‌ها ۱۰ نفر است.

(۳) جلسات بازبینی به دلیل شرکت‌کنندگان پر تعداد آن بیشتر از ۳ ساعت به طول می‌انجامند.

(۴) هیچ کدام مناسب نیستند

۳۰- کدام یک از اتفاقات زیر باعث افزایش رسمیت جلسات بازبینی نمی‌شود؟

- (۱) تعریف شدن نقش‌های مجزا برای بازبین‌ها
- (۲) حضور مدیران ارشد پروژه و کارفرما در جلسه
- (۳) برنامه‌ریزی و آماده‌شدن برای جلسات بازبینی به مقدار کافی
- (۴) اقداماتی توسط بازبین‌ها در راستای پیگیری انجام تصحیحاتی که در جلسه درباره‌ی آن‌ها تصمیم گرفته شده است.

خسته نباشید ...