

به نام خدا

میانترم دوم

مهندسی نرم افزار – نیمسال اول ۴۰۲

➤ تاریخ آزمون: پنجشنبه ۱۴ / ۱۰ / ۴۰۲

➤ زمان شروع: ۹:۰۰ صبح

➤ زمان پایان: ۱۴:۰۰ ظهر



دانشکده مهندسی کامپیوتر – دانشگاه شریف

مدرس: دکتر مهران ریواده

بخش اول – سوالات تستی

۱- کدام یک از موارد زیر در رابطه با انتزاع (Abstraction) در طراحی نرم افزار درست نیست؟

- (a) انتزاع یکی از روش های اساسی برای ساده کردن سیستم ها یا مولفه های (Component) پیچیده است.
- (b) انتزاع سطوح متفاوتی دارد و هر چقدر از سطوح بالاتر به سطوح پایین تر برویم، جزئیات راه حل ارائه شده مساله، پنهان می ماند.
- (c) انتزاع رویه ای (Procedural Abstraction) دنباله ای از دستورالعمل ها است که عملکرد محدود و مشخصی دارند. برای مثال افزایش سرعت ماشین.
- (d) انتزاع داده ای (Data Abstraction) یک مجموعه نام گذاری شده از داده ها است که یک شی داده ای را تعریف می کند. مانند یک ماشین.

۲- بر اساس اصول طراحی رابط کاربری Mandel برای کاهش بار حافظه کاربر، کدام یک از گزینه های زیر نادرست است؟

- (a) نشانه های بصری: رابط باید شامل نشانه های بصری برای کمک به کاربران در شناسایی اقدامات و ورودی های گذشته باشد تا بار حافظه کوتاه مدت آن ها کاهش یابد.
- (b) پیش فرض های معنادار: تنظیمات پیش فرض اولیه باید برای کاربر متوسط منطقی باشد، با انعطاف پذیری برای سفارشی سازی فردی و گزینه ای برای بازنشانی به تنظیمات پیش فرض اصلی.
- (c) میانبرهای شهودی: طراحی رابط باید شامل میانبرهای شهودی، مانند میانبرهای حافظه که مستقیماً به عملکردهای خود مرتبط هستند، برای سهولت در به خاطر سپردن باشد.
- (d) نمایش اولیه جزئیات: در حالی که اطلاعات به صورت سلسله مراتبی سازماندهی می شود، رابط باید ابتدا سطح جامعی از جزئیات را برای ارائه زمینه کامل به کاربران قبل از ورود به وظایف خاص نمایش دهد.

۳- یک طراح اپلیکیشن های موبایل، در یکی از پروژه های خود برای یک دکمه از اپلیکیشن در دو صفحه متفاوت، نام گذاری متفاوتی قرار داده است در صورتی که هر دو دکمه یک عملکرد (Functionality) دارند، این طراح کدام یک از اشتباهات رایج در طراحی UI را انجام داده است؟

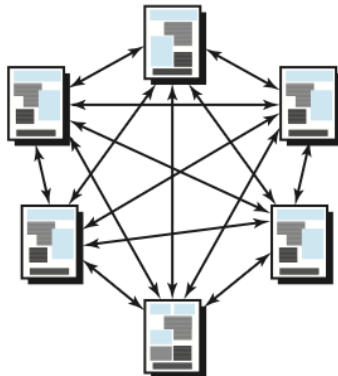
- (a) Overdesigning
- (b) Verbiage
- (c) Nonstandard Interaction
- (d) Inconsistency

۴- کدام یک از جملات زیر به درستی یکی از اشتباهات رایجی را که هنگام استفاده از طراحی مبتنی بر الگو در توسعه نرم افزار رخ می دهد، توصیف می کند؟

- (a) الگوها در طراحی نرم افزار همیشه بی نقص هستند و نیازی به تطبیق برای فضاهای مشکل خاص ندارند.
- (b) جستجو برای نظر شخص دوم یا بازبینی کار طراحی توصیه نمی شود زیرا ممکن است باعث سردرگمی و پیچیدگی در فرآیند طراحی شود.

- (c) یکی از اشتباهات رایج انتخاب الگو بدون درک کامل از مشکل و زمینه (context) مشکل می باشد که باعث می شود الگوی انتخاب شده، به صورت کامل به همه نیازها پاسخ ندهد.
- (d) طراحی مبتنی بر الگو به طور ذاتی تمام پارامترها و زمینه های (context) احتمالی یک مشکل را در نظر می گیرد، بنابراین تحلیل عمیق مشکل قبل از انتخاب الگو غیر ضروری است.

۵- کدام گزینه در مورد ساختار زیر که به آن Pure web هم گفته می شود، برای Web App ها صحیح است؟



- (a) از جهات بسیاری شبیه معماری هایی است که برای سیستم های شیء گرا وجود دارند.
- (b) در این ساختار، هر صفحه وب می تواند کنترل را به هر صفحه دیگری بسپارد.
- (c) این ساختار گرچه انعطاف پذیری قابل توجهی در Navigation را فراهم می کند اما برای کاربر گیج کننده است.
- (d) همه موارد صحیح هستند.

بخش دوم – سوالات تشریحی

۶- چه تفاوتی بین الگوهای معماری با Style های معماری وجود دارد؟ مختصراً شرح دهید.

۷- سناریویی را در نظر بگیرید که در آن کاربر در حال تعامل با یک برنامه موبایل جدید است که برای مدیریت امور مالی شخصی طراحی شده است. این برنامه به کاربران امکان می دهد هزینه ها را پیگیری کنند، بودجه را تنظیم کنند و گزارش های مالی را مشاهده کنند. با این حال، کاربران برخی از مشکلات را هنگام استفاده از برنامه گزارش کرده اند. بر اساس اصول طراحی Bruce Tognozzi، مشخص کنید کدام اصل(ها) ممکن است در این سناریو نقض شده باشد و دلایل آن را بیان کنید.

مشکلات گزارش شده:

- برنامه اقدامات مربوطه را پیشنهاد نمی کند یا مراحل بعدی کاربر را پیش بینی نمی کند، مانند پیشنهاد تنظیم بودجه بر اساس الگوهای هزینه های گذشته.
- رابط برنامه با عملکردهای بیش از حد در صفحه اصلی به هم ریخته است، که تمرکز روی یک کار واحد مانند وارد کردن هزینه های روزانه را دشوار می کند.
- همینطور کاربران جدید گزارش کرده اند که درک نحوه پیمایش در برنامه و استفاده از ویژگی های آن مشکل دارند.
- ساختار Navigation گیج کننده است بطوریکه برخی از عملکردها که در زیر چندین لایه از منوها مدفون شده اند و یافتن آن ها را سخت می کند.

۸- چه زمانی از Component Wrapping استفاده می کنیم؟ تکنیک های مورد استفاده در آن را مختصر توضیح دهید.

۹- سناریوهای زیر را در نظر بگیرید. هر سناریو ممکن است اصول طراحی نرم افزار شامل اصل OCP، اصل LSP، اصل DIP، اصل ISP را نقض کرده باشند و یا دارای Coupling بالا یا Cohesion پایین باشند. بررسی نمایید هر یک از این سناریوها چه مشکلی دارند و چرا؟

- یک کلاس Animal متدی به نام makeSound دارد. کلاس Dog و کلاس Cat هر دو از Animal به ارث می برند. نوع جدیدی از حیوانات به نام Fish اضافه می شود، ولی متد makeSound برای آن استفاده نمی شود.
- یک کلاس رابط کاربری مسئول مدیریت ورودی های ماوس، ورودی های صفحه کلید، رندر کردن گرافیک و مدیریت پیام های شبکه است.
- کلاس دسترسی به پایگاه داده یک سیستم نرم افزاری از مصرف کنندگان می خواهد که مدیریت تراکنش، مدیریت اتصال و مدیریت خطا را پیاده سازی کنند، حتی اگر فقط به اجرای یک عملیات خواندن ساده نیاز داشته باشند.
- یک سیستم پردازش پرداخت به یک درگاه پرداخت خاص وابسته است و هر تغییری که در درگاه پرداخت ایجاد شود، تأثیر مستقیمی بر سیستم پردازش پرداخت خواهد داشت.

۱۰- در طراحی Pattern-Based زمانی که تعداد design patternهایی که می خواهید از بین آنها انتخاب کنید زیاد می شود، مرتب سازی به یک ضرورت تبدیل می شود. چه روشی برای این مرتب سازی و انتخاب پیشنهاد می کنید؟ شکل کلی روش خود را توضیح دهید.

موفق باشید

تیم آموزش مهندسی نرم افزار