

فصل ۱۵: طراحی واسط کاربر

طراحی واسط بر سه حوزه موضوع مهم به شرح زیر تأکید دارد:

۱. طراحی واسط بین اجزای نرم افزار،
 ۲. طراحی واسطها بین نرم افزار و سایر تولیدکنندگان و مصرف کنندگان اطلاعاتی غیر بشری (یعنی سایر اشیاء خارجی) و
 ۳. طراحی واسط بین یک انسان (یعنی کاربر) و کامپیوتر.
- در این فصل صرفاً بر سومین مقوله طراحی واسطها یعنی طراحی واسط کاربر توجه خواهیم داشت.

قواعد طلایی

Mendal در کتاب خود با عنوان طراحی واسط، [MAN 97] سه "قانون طلایی" را به صورت زیر ارایه می کند:

- ۱- واگذاری کنترل به کاربر
- ۲- کاهش بار حافظه کاربر
- ۳- سازگار کردن واسطها

این قوانین طلایی، عملاً مبنایی برای مجموعه اصول طراحی واسط کاربر هستند که این فعالیت مهم طراحی نرم افزاری را هدایت می کنند. پ

واگذاری کنترل به کاربر

در این قانون رعایت نکات زیر الزامی است.

- ۱- تعیین شیوه های تعاملی به نحوی که کاربر را مجبور به اعمال غیر ضروری یا نامطلوب نکند.
- ۲- ایجاد تعامل انعطاف پذیر در ارتباط با کاربر
- ۳- امکان ایجاد وقفه و خنثی سازی (بازگشت) در تعامل کاربر.
- ۴- کارآمد ساختن تعامل همراه با پیشرفت سطوح مهارتی و امکان سفارشی کردن آن. کاربران اغلب در می یابند که زنجیره یکسانی از تعاملات را به طور مکرر انجام می دهند. طراحی یک ماکرو، مکانیزی که به کاربر پیشرفته امکان می دهد برای تسهیل تعامل، واسط را سفارشی کند، ارزشمند و مفید است.
- ۵- مخفی کردن موارد فنی داخلی از کاربران عادی. واسط کاربر بایستی او را به درون عالم واقعی برنامه کاربردی سوق دهد. کاربر نباید از سیستم عامل، وظایف مدیریت فایل یا سایر فناوری ها و ... مطلع باشد.
- ۶- طراحی تعامل مستقیم با اشیایی که روی صفحه نمایش ظاهر می شوند. کاربر با دستکاری اشیایی که برای انجام یک عمل ضروری هستند، احساس کنترل می کند، درست مثل زمانی که آن شیء یک شیء فیزیکی است. مثلاً واسط کاربری که امکان کشیدن یک شیء را (یعنی تغییر مقیاس آن از لحاظ سایر) به کاربر می دهد، نمونه ای از دستکاری مستقیم است.

کاستن از بار حافظه کاربر

در این قانون نیز رعایت نکات زیر الزامی است.

۱. کاهش بار در حافظه کوتاه مدت. هنگامی که کاربران درگیر وظایف پیچیده هستند، باید بار حافظه کوتاه مدت را کاهش دهند. این عمل با ایجاد علایم بصری میسر است که به کاربر امکان می دهند تا کارهای قبلی را شناسایی کنند، نه این که مجبور باشد آنها را به یاد آورد.

۲. ایجاد پیش‌گزیده‌های معنی‌دار. مجموعه آغازین پیش‌فرض‌ها باید برای کاربر متوسط معنی‌دار باشد، اما کاربر باید بتواند اولویت‌های فردی خود را مشخص کند. هر چند که امکان تنظیم مجدد، بایستی موجود باشد تا تعریف مجدد مقادیر اصلی پیش‌گزیده امکان‌پذیر گردد.
- ۴- تعیین میان‌برهایی که شهودی هستند. زمانی که برای انجام عملکرد سیستم از مجموعه‌ای از کلمات استفاده می‌شود، (مثلاً $P - alt$ برای فعال کردن عمل چاپ)، کلمات حفظی باید به شیوه‌ای که به خاطر آوردن آن آسان باشد و به عمل مورد نظر مرتبط گردد. (به عنوان مثال، حرف اول آن عمل، فراخوانده شود).
- ۵- طرح بصری واسط باید براساس استعاره جهان واقعی باشد. به عنوان مثال سیستم پرداخت فاکتور باید برای هدایت کاربر طی فرایند پرداخت صورت‌حساب، از استعاره دسته چک و ثبت چک استفاده کند. این مساله به کاربر امکان می‌دهد تا به جای حفظ سلسله کارهای غیر متعارف تعاملی، به علایم بصری شناخته شده متوسل شود.
- ۶- آشکارسازی اطلاعات به شیوه‌ای تدریجی. واسط باید به طور سلسله مراتبی سازمان‌دهی شود. یعنی آن که اطلاعات درباره یک عمل، شی یا یک شیوه باید ابتدا در سطح بالایی از انتزاع آرایه گردد. جزئیات بیشتر باید پس از اعلام علاقه کاربر با انتخاب وی از طریق ماوس، در اختیار او قرار گیرد. مثال رایج در بسیاری از برنامه‌های کاربردی واژه‌پردازی، عمل خط زیر است. این کارکرد یکی از چندین کارکردی است که در منوی قالب‌بندی متن قرار دارد. هر چند که تمامی امکانات خط زیر، فهرست نمی‌شوند. کاربر باید ابتدا خط زیر را انتخاب کند و سپس تمامی امکانات خط زیر (خط زیر تک خطی، دو خطی و نقطه‌چین) به نمایش درمی‌آیند.

سازگاری واسط

در این قانون نیز رعایت نکات زیر الزامی است.

- ۱- قراردادن عمل فعلی در یک بافت معنی‌دار توسط کاربر. بسیاری از واسط‌ها، لایه‌های پیچیده تعاملی را با تصاویر زیادی در صفحه نمایش پیاده می‌کنند. تهیه نشان‌گرها (مثل عناوین پنجره، شمایل‌های گرافیکی، کدگذاری ثابت رنگ)، از این نظر که کاربر را قادر می‌سازند تا محیط کاری موجود را بشناسد، اهمیت دارند. به علاوه، کاربر باید بتواند مبدأ خود و امکانات موجود برای گذر به عملی جدید را تعیین کند.
- ۲- حفظ ثبات در خانواده برنامه‌های کاربردی. مجموعه‌ای از برنامه‌های کاربردی (یا محصولات) باید همگی قوانین یکسان طراحی را پیاده سازی کنند، به نحوی که سازگاری در تمامی تعاملات و محاوره‌ها حفظ شود.
- ۳- اگر مدل‌های تعاملی پیشین انتظاراتی را در کاربر به وجود آورده‌اند، تا زمانی که دلیل قانع‌کننده‌ای نداشتند از انجام تغییرات خودداری کنید. پس از تبدیل یک ترتیب خاص تعاملی و یا یک استاندارد عملی (مثل کاربرد $S - alt$ برای ذخیره فایل) از تغییر آن خودداری کنید، زیرا کاربر در مواجهه با هر برنامه کاربردی دیگر همین انتظار را دارد و انجام یک تغییر (مثل کاربرد $S - alt$ برای تغییر مقیاس) سبب آشفتگی و سردرگمی او خواهد شد.

اصول طراحی واسط که در این بخش و قسمت‌های قبلی مورد بحث قرار گرفتند، راهنمای اصلی یک مهندس نرم‌افزار به شمار می‌روند. در قسمت‌های بعدی، فرآیند طراحی واسط را بررسی خواهیم کرد.

طراحی واسط کاربر

فرآیند کلی طراحی واسط کاربر، با ایجاد مدل‌های مختلف کارکرد سیستم (آن طور که از بیرون مشاهده می‌شود) آغاز می‌گردد. سپس وظایف انسانی و کامپیوتری لازم برای تحقق کارکرد سیستم، توصیف می‌شوند، موضوعات طراحی

که در تمام طراحی‌های واسط کاربرد دارند مدنظر قرار می‌گیرند، برای الگوسازی و پیاده‌سازی نهایی مدل طراحی، ابزارهایی به کار می‌روند و نتیجه از لحاظ کیفی ارزیابی می‌گردد.

هنگام طراحی محیط تعامل از طریق تایپ فرمانها به مسایل زیر باید توجه داشت:

- آیا هر یک از گزینه‌های منو دارای یک فرمان متناظر هست؟
- فرمانها چه شکلی به خود می‌گیرند؟
- فراگیری و به خاطر سپردن فرمانها چقدر دشوار خواهد بود؟ اگر فرمانی فراموش شد، چه می‌توان کرد؟
- آیا کاربر می‌تواند فرمانها را به سلیقه خویش مختصر و کوتاه کند؟

مدل‌های طراحی واسط

به هنگام طراحی یک واسط کاربر، چهار مدل مختلف به کار می‌آیند. مهندس نرم‌افزار **مدل طراحی** را ایجاد می‌کند، مهندس فاکتورهای انسانی (یا مهندس نرم‌افزار) **مدل کاربر** را تعیین می‌کند، کاربر نهایی یک تصویر ذهنی می‌سازد که غالباً **مدل ذهنی کاربر** را به وجود می‌آورند. پیاده کنندگان سیستم نیز یک **تصویر سیستم** ایجاد می‌کنند. متأسفانه هر یک از این مدل‌ها ممکن است تفاوت قابل ملاحظه‌ای با یکدیگر داشته باشند. نقش طراح واسط، رفع اختلافات و به دست دادن یک نمایش منسجم و سازگار از واسط است.

مدل طراحی کل سیستم، تلفیقی از نمایش داده‌ها، معماری، واسط و بازنمایی رویه‌ای نرم‌افزار می‌باشد. تعیین نیازها ممکن است محدودیت‌های خاصی را مطرح کند که به تعیین کاربر سیستم کمک می‌کنند. اما طراحی واسط، اغلب تنها لازمه مدل طراحی است.

مدل کاربر، نمایی از کاربران نهایی سیستم را ترسیم می‌کند. برای ساخت یک رابط کار مؤثر، "تمام کار طراحی باید با درک درستی از کاربران مورد نظر، از جمله مشخصات سن، جنسیت، توانایی‌های جسمی، سابقه تحصیلی، فرهنگی یا قومی، انگیزه، اهداف و شخصیت آنها، آغاز گردد. به علاوه، کاربران را می‌توان در گروه‌های زیر طبقه‌بندی کرد:

۱- **کاربران مبتدی**. از دانش نحوی سیستم برخوردار نیستند و دانش معنایی آنها از برنامه کاربردی یا کاربرد کامپیوتر به طور کلی، اندک است.

۲- **کاربران مطلع و دوره‌ای**. دانش معنایی معقول از برنامه کاربردی دارند اما به یادآوری نسبتاً کم دانش نحوی لازم برای کاربرد رابط دارند.

۳- **کاربران مطلع و دائمی**. دانش نحوی و معنایی مناسب دارند که اغلب به "مشخصه کاربر ماهر" منجر می‌شود، یعنی کاربرانی که به دنبال میان‌برها و حالت‌های اختصاری تعامل هستند.

ادراک سیستم (مدل ذهنی کاربر). تصویری از سیستم است که کاربر نهایی در ذهن خود ایجاد می‌کند. مثلاً اگر از کاربر یک واژه‌پرداز پرداز خاص بخواهیم تا کارکرد آن را توصیف کند، پاسخ او براساس درک او از سیستم خواهد بود. صحت توصیف بستگی به شرح حال کاربر (به عنوان مثال جواب کاربران مبتدی در نهایت مجمل و ناقص است) و آشنایی کلی با نرم‌افزار در حیطه برنامه کاربردی دارد.

کاربری که درک کاملی از واژه‌پردازها دارد. اما تنها با واژه‌پرداز خاصی کار کرده است، در عمل نسبت به تازه‌کاری که هفته‌ها وقت صرف یادگیری سیستم نموده ممکن است توصیف کامل‌تر و جامع‌تری را ارائه دهد.

تصویر سیستم، ترکیبی از نمود بیرونی سیستم کامپیوتری (یعنی ظاهر و عملکرد رابط) به همراه تمامی اطلاعات پشتیبان (کتاب‌ها، دستنویس‌ها، نوارهای ویدیویی و فایل‌ها راهنما) است که نحو و معنا شناسایی سیستم را توصیف می‌کنند. زمانی که تصویر سیستم و درک سیستم یکسان باشند، عموماً کاربران با نرم‌افزار احساس راحتی نموده و به طور مؤثر آن را به کار می‌برند. به منظور ادغام مدل‌ها، مدل طراحی باید سازگاری با اطلاعات موجود در مدل کاربر توسعه یافته باشد و تصویر سیستم اطلاعات نحوی و معنایی درباره واسط را دقیقاً منعکس کند.

چند مسأله طراحی راهنماگام پرداختن به تسهیلات راهنما باید در نظر داشت:

- آیا راهنما برای کلیه عملکردهای سیستم و در همه اوقات تعامل با سیستم در دسترس خواهد بود؟
- کاربر چگونه درخواست کمک و راهنمایی می کند؟
- راهنما چگونه ارایه خواهد شد؟
- کاربر چگونه به تعاملهای عادی باز می گردد؟
- اطلاعات راهنما چگونه ساختاری دارند؟

فرایند طراحی واسط کاربر

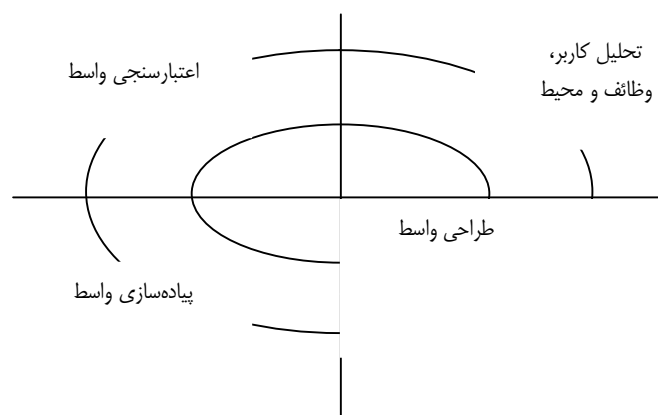
فرایند طراحی واسطهای کاربر، تکراری است و با استفاده از مدل حلزونی، مشابه آن چه در فصل ۲ مورد بحث قرار گرفت، قابل ارایه است. با مراجعه به شکل ۱، روند طراحی واسط کاربر، چهار فعالیت مجزای ساختاری را دربردارد.

۱- تحلیل و الگوسازی کاربر، وظیفه و محیط و مدل سازی

۲- طراحی واسط

۳- ساخت رابط

۴- اعتبارسنجی واسط



شکل ۱: فرایند طراحی واسط کاربری

سیستم توسعه واسط کاربر (UIDS) با استفاده از مؤلفه های نرم افزاری، راهکاری برای موارد زیر فراهم می آورد:

- مدیریت دستگاه های ورودی (مثل موس یا صفحه کلید)
- اعتبار سنجی ورودی کاربر
- کنترل خطا و نمایش پیامهای خطا
- فراهم آوردن باز خود (مثل بازتاب ورودی خودکار)
- فراهم آوردن راهنما و پیام
- کنترل پنجره ها و فیلدها، حرکت در داخل پنجره ها
- برقراری ارتباط میان نرم افزار کاربردی و واسط
- جدا کردن برنامه کاربردی از عملکردهای مدیریت واسط
- مجاز کردن کاربر به سفارشی کردن واسط

بنابراین در طراحی واسط کاربری باید مراحل زیر طی گردد:

- تعیین اهداف و مقاصد هر وظیفه
- نگاشت هر هدف/نیت در تعدادی عملیات مشخص
- مشخص کردن دنباله عملیاتی وظایف اصلی و فرعی، که چون در سطح واسط اجرا می شوند، سناریوی کاربر نیز خوانده می شوند
- نشان دادن حالت سیستم
- تعریف راهکارهای کنترلی
- نشان دادن چگونگی تاثیرپذیرفتن حالت سیستم از راهکارهای کنترلی
- نشان دادن چگونگی تفسیر حالت سیستم توسط کاربر با استفاده از اطلاعات به دست آمده از طریق واسط

مسایل طراحی واسط ها

در حین تکمیل طراحی واسط کاربر، چهار مساله معمول طراحی تقریباً همیشه سطحی تلقی می شوند:

زمان پاسخ گویی سیستم، تسهیلات کمکی کاربر، خطاگردانی اطلاعات و برچسب گذاری فرمان.

متأسفانه، بسیاری از طراحان کمی دیر این مسایل را در فرآیند طراحی مدنظر قرار می دهند (گاهی قبل از در دسترس قرار گرفتن یک نمونه عملی، اشاره مختصر به یک مشکل صورت نمی گیرد). تکرار غیر ضروری، تأخیرهای پروژه و نارضایتی مشتری، اغلب از پیامدهای حاصله است. پس بهتر آن است که هر یک از مسایل در آغاز طراحی نرم افزار و به هنگام انجام راحت تغییرات و پایین بودن هزینه ها، مورد توجه قرار گیرند.

زمان پاسخ گویی سیستم از زمانی که کاربر عمل کنترلی را انجام می دهد (مثلاً کلید بازگشت را زده یا روی ماوس کلیک می کند) تا زمان پاسخ گویی نرم افزار با اقدام یا خروجی مطلوب، اندازه گیری می شود.

زمان پاسخ گویی سیستم دو ویژگی مهم دارد: **طول و تغییرپذیری**. اگر **طول** پاسخ گویی سیستم بسیار طولانی باشد، ناامیدی و فشار روی کاربر، نتیجه ای اجتناب ناپذیر است. هر چند که اگر واسط، باعث دستپاچی کاربر شود، زمان پاسخ گویی بسیار کوتاه نیز می تواند مضر باشد. پاسخ گویی سریع ممکن است موجب عجله کاربر و بنابراین انجام اشتباه از سوی او شود.

تغییرپذیری به انحراف از زمان میانگین پاسخ گویی اشاره داشته و از خیلی جهات، مهم ترین مشخصه زمان پاسخ گویی به شمار می رود. تغییرپذیری کم، حتی در صورت طولانی بودن زمان پاسخ گویی، به کاربر امکان می دهد تا به طریق مناسب تعامل را برقرار کند. به عنوان مثال، پاسخ یک ثانیه ای به یک فرمان، به پاسخ گویی متغیر بین یک تا دو و نیم ثانیه، ترجیح دارد. کاربر همیشه بلامتکلیف است و نمی تواند که در پشت صحنه چه خبر است و چه اتفاقی افتاده است.

تقریباً تمامی کاربران یک سیستم تعاملی کامپیوتری، گهگاه به کمک نیاز دارند. دو نوع امکانات کمکی عبارتند از: **یکپارچه و افزودنی**، [RUB 88] **تسهیلات کمکی یکپارچه** از آغاز در داخل نرم افزار طراحی می شود. این نوع کمک به کاربر غالباً حساس به متن است و کاربر را قادر می سازد تا از میان موضوعات مرتبط با اعمال در حال اجرا، اقدام به انتخاب کند. واضح است که این امر، زمان لازم برای دریافت کمک توسط کاربر را کاهش داده و کاربر پسندی واسط را افزایش می دهد. **تسهیلات کمکی افزودنی**. پس از ساخت سیستم به نرم افزار افزوده می شود. ممکن است کاربر مجبور شود برای یافتن راهنمایی صحیح و مناسب، فهرستی با صدها موضوع را جستجو کند و اغلب با شروع نادرست، اطلاعات غیرمرتبط را دریافت نماید. شکی نیست که امکانات کمکی **یکپارچه** بر نوع **افزودنی** آن برتری دارد.

هنگام بروز خطا، پیغام های خطا و هشدارها. "اخبار بدی" است که به کاربران سیستم های تعاملی ارایه می گردد. در بدترین حالت، پیغام های خطا و اخطارها، اطلاعات بی فایده یا گمراه کننده را منتقل کرده و تنها باعث تشدید ناکامی کاربر می شوند. تعداد کاربران کامپیوتر که با خطایی از نوع زیر مواجه نشده باشند، بسیار اندک است.

فرمان تایپ شده زمانی، رایج ترین شیوه محاوره بین کاربر و نرم افزار سیستم بود و در هر نوع برنامه کاربردی به طور معمول به کار می رفت. امروزه، استفاده از رابط های پنجره ای و اشاره و انتخاب، کاربرد فرامین تایپ شده را کاهش داده، اما بسیاری از کاربران ماهر همچنان شیوه ارتباطی مجهز به فرمان را ترجیح می دهند. به هنگام انتخاب فرامین تایپ شده به عنوان نوعی شیوه محاوره، برخی مسائل طراحی مورد توجه قرار می گیرند:

- ♦ آیا هر یک از انتخاب های منو یک فرمان مرتبط خواهد داشت؟
- ♦ فرامین به چه شکلی خواهند بود؟ امکانات موجود عبارتند از: توالی کنترل (مثل Alt + P)؛ کلیدهای تابعی، واژه تایپ شده.
- ♦ یادگیری و به خاطر سپردن فرامین تا چه حد دشوار است؟ در صورت فراموشی یک فرمان، چه می توان کرد؟
- ♦ آیا می توان فرامین را توسط کاربر سفارشی یا اختصاری کرد؟

تحلیل محیط کاربر بر محیط کار فیزیکی تکیه دارد. از جمله پرسش هایی که باید مطرح شوند، عبارتند از:

- واسط از نظر فیزیکی در کجا قرار داده خواهد شد؟
- آیا کاربر به حالت نشسته کار می کند یا ایستاده، یا کارهای دیگری انجام می دهد که با واسط بی ارتباط هستند؟
- آیا سخت افزار واسط محدودیتهای جا، نور و سرو صدا را در بر می گیرد؟

به طور کلی هر پیام خطا یا هشدار تولید شده توسط یک سیستم محاوره ای باید دارای ویژگیهای زیر باشد:

- پیام باید مشکل را به زبانی شرح دهد که کار بر قادر به درک آن باشد
- پیام باید حاوی یک توصیه سازنده برای رهایی از وضعیت خطا باشد
- پیام باید هر گونه تبعات منفی خطا (مثلا فایل های داده ای مخدوش شده) را خاطر نشان کند تا کاربر بتواند آنها را چک کند
- پیام باید با یک نشانه سمعی یا بصری همراه باشد
- پیام باید قضاوت گونه باشد

ارزیابی طراحی

پس از ایجاد الگوی عملی واسط کاربر، این مدل باید مورد ارزیابی قرار گیرد تا معلوم شود که آیا نیازهای کاربر را برطرف می کند یا خیر؟ ارزیابی می تواند در یک طیف رسمی صورت گیرد که گستره آن با انجام آزمونی غیر رسمی که ضمن آن کاربر بازتابی بدون فکر قبلی دارد شروع شده و به مطالعه رسمی ختم می شود که از روش های آماری برای ارزیابی پرسش نامه های تکمیل شده توسط کاربران نهایی، استفاده می کند. چرخه ارزیابی واسط کاربر مشبیه شکل ۲ می باشد.

برخی معیارهای ارزیابی [MOR81] را هنگام بررسی های اولیه طراحی، می توان اعمال کرد:

- ۱- طول و پیچیدگی مشخصات مکتوب سیستم و واسط آن، بیانگر میزان یادگیری لازم توسط کاربران سیستم می باشد.
- ۲- تعداد وظایف تعیین شده کاربر و میانگین اعمال در هر کار، نشان دهنده زمان محاوره و کارایی کلی سیستم است.
- ۳- تعداد اعمال، وظایف و وضعیت های سیستم که در مدل طراحی تعیین شده، به بار حافظه کاربران سیستم دلالت دارد.

۴- روش ارتباط واسط، امکانات کمکی و خطاگردانی، در کل بیانگر پیچیدگی واسط و میزان پذیرش از سوی کاربر می باشد.

پس از ساخته شدن اولین مدل، طراحی می تواند مجموعه ای از داده های کیفی و کمی را که به ارزیابی واسط کمک خواهند کرد، را جمع آوری می کند. ویژگی های پرسشنامه هایی که می توان بین کاربران توزیع کرد می تواند شامل موارد زیر باشد:

۱. پاسخ های ساده آری/ خیر داشته باشد.

۲. پاسخ عددی داشته باشد.

۳. پاسخ مقیاسی داشته باشد.

۴. پاسخ درصدی داشته باشد.

سوالات می توانند به صورت زیر باشند:

۱. آیا نمادها (ICONS) خود توصیف هستند؟ اگر نیستند کدام نماد نیاز به توصیف دارد ؟

۲. آیا عملیات را به آسانی می توان به خاطر سپرد و اجرا کرد؟

۳. تا کنون از چند عمل متفاوت استفاده کرده اید؟

۴. سهولت فراگیری عملیات اصلی سیستم تا چه حدی است؟ (بین ۱ تا ۵)

۵. در مقایسه با واسط های دیگری که استفاده کرده اید، به این واسط چه امتیازی می دهید؟

پس از جمع آوری اطلاعات مربوطه باید نسبت به تحلیل و جمع بندی آن اقدام نمود.

تست های فصل ۱۵ : طراحی واسط ها

۱. سه قاعده طلایی مندل (mandel) در طراحی واسطها شامل ۱- سپردن کنترل به کاربر ۲- کاستن از بار حافظه کاربر و سازگار ساختن واسطها است.
(الف) درست (ب) غلط
۲. واسطهای خارجی در تعامل با موجودیت های بیرونی سیستم در مدل تحلیل با شناسایی داده و کنترل مورد نیاز طراحی می گردد.
(الف) درست (ب) غلط
۳. در طراحی واسطهای کاربر باید نحوه پیاده سازی واسط، محیط (فناوری) مورد استفاده و عوامل حساس کاربر را در نظر گرفت.
(الف) درست (ب) غلط
۴. کاربر در زمان کار با سیستم، زمان پاسخگویی سیستم را مورد ارزیابی قرار می دهد. معیارهای قابل اندازه گیری برای این کار چیست؟
(الف) میانگین طول زمان پاسخ (ب) طول زمان پاسخ (ج) طول زمان پاسخ و تغییرپذیری (د) انحرافات زمان پاسخ
۵. واسطهای کاربر باید با توجه به فرهنگ و دانش عمومی کاربران تهیه گردد.
(الف) درست (ب) غلط
۶. کدام یک از اصول طراحی واسط زیر اجازه نمی دهد که کاربر در تعامل کنترلی با کامپیوتر قرار داشته باشد.
(الف) تعاملی که بتواند اجرای برنامه را قطع نماید.
(ب) تعاملی که بتواند عملیات را بازگردانی نماید.
(ج) پنهان کردن امور فنی داخلی از دید کاربر
(د) فقط شامل یک روش تعریف دشوار و سخت برای تکمیل یک وظیفه
۷. کدام یک از اصول طراحی واسط زیر، بار حافظه کاربر را کاهش می دهد.
(الف) تعریف Shortcut های مستقیم
(ب) فاش کردن اطلاعات به شیوه ای فزاینده
(ج) استقرار پیش فرض های با معنی
(د) تمام موارد فوق
۸. دلیل کاستن از بار حافظه کاربر آن است که وی بتواند در تعامل با کامپیوتر سریعتر کار را تمام کند.
(الف) درست (ب) نادرست
۹. سازگاری واسطها منجر به
(الف) مکانیزم های ورودی در تمام نرم افزار کاربردی یکسان باقی می ماند.
(ب) هر برنامه کاربردی باید نگاه و احساس منحصر به خود را داشته باشد.
(ج) روش های راهبردی حساس به محتوا هستند.
(د) موارد الف و ب
۱۰. اگر مدلهای تعاملی گذشته باری توقعات یک کاربر خاص ساخته شده باشند به طور کلی تغییر مدل ها ایده خوبی نیست.
(الف) درست (ب) نادرست
۱۱. کدام یک از مدل های زیر سودمندی کاربر نهایی یک سیستم کامپیوتری را شرح می دهد؟
(الف) مدل طراحی (ب) مدل کاربری (ج) مدل مربوط به کاربر (برداشت سیستمی) (د) تصویر سیستمی
۱۲. کدام یک از مدل های زیر تصویری از یک سیستم را ارائه می دهد که کاربر نهایی در ذهن خود ایجاد کرده است؟
(الف) مدل طراحی (ب) مدل کاربری (ج) برداشت سیستمی (د) تصویر سیستمی

۱۳. کدام یک از مدل های زیر نگاه و نگرشی به واسط کاربر با پشتیبانی تمام اطلاعات را دارد؟

الف) مدل کاربری ب) مدل کاربر ج) برداشت سیستمی د) تصویر سیستمی

۱۴. کدام یک از فعالیت های چارچوبه ای (پشتیبانی) زیر به طور طبیعی در فرآیند طراحی واسط مورد استفاده

قرار نمی گیرند:

الف) برآورد هزینه ب) ساخت واسط ج) اعتبارسنجی واسط د) تحلیل کاربر وظایف

۱۵. کدام نگرش بر تحلیل وظایف کاربر در طراحی واسط کاربر مفید است؟

الف) داشتن کاربرانی که ترجیحات خود را توسط پرسشنامه بیان کرده اند.

ب) تکیه بر قضاوت برنامه نویسان با تجربه

ج) مطالعه سیستم های مکانیزه شده مرتبط

د) مشاهده نحوه انجام کار دستی کاربران