

رابط كاربر القايى



رابط کاربر القایی

خلاصه: این مقاله نمونه‌ی جدید رابط کاربر را که رابط کاربر القایی (Inductive User Interface) یا هدایت کننده‌ی القایی نامیده می‌شود، شرح خواهد داد. این گونه رابط با استفاده از صفحات مجزاً برای مقاصد مختلف باعث سادگی و وضوح عملکرد برنامه می‌شود.

اشکال مختلف رابط کاربر القایی: اصولی که در این مقاله مورد بحث قرار گرفته‌اند، فارق از نوع شیء، ساختار طراحی و عناصر تصویری هستند و فضای گسترده‌ای را برای طراحی و انعطاف‌پذیری باقی می‌گذارند.

رابط کاربر القایی چیست؟

IUI گونه‌ای جدید از رابط کاربر است که با هدف کارایی بیشتر نرم‌افزار و سهولت فراگیری آن برای کاربر طراحی شده است. مایکروسافت این نوع رابط کاربر را در نرم‌افزار تجاری و عظیم Money 2000 با موفقیت آزمایش کرده است. در این مدل کاربران به آسانی قادر هستند تمامی اعمالی را که در رابط‌های سنتی انجام می‌داده‌اند، تکرار کنند و علاوه بر آن، امکانات جدید و مورد نیاز خود را بیابند. در بسیاری از نرم‌افزارهای تجاری از رابط‌های کاربری که متشکل از یک سری شیء است استفاده شده است و تصمیم‌گیری درباره‌ی هدف هر بخش و پیدا کردن راه حل مشکلات به کاربر واگذار می‌شود.

نکته‌ی مهم: IUI در ابتدای مسیر قرار دارد و تا کشف بهترین راهها برای بهترین کارایی مسیری طولانی در پیش است. در این مقاله نیز فقط به شرح کلیات خواهیم پرداخت و از معرفی راهبردهایی خاص خودداری خواهیم کرد.

IUI در عمل: حل یک مسئله‌ی معمول در طراحی

این بخش به یک مشکل متداول در طراحی نرم‌افزار می‌پردازد و نشان خواهد داد که چگونه می‌توان با استفاده از رابط کاربر القایی بر آن غلبه کرد.

مسئله: کار کردن با نرم‌افزار مشکل است.

کار کردن با بسیاری از نرم‌افزارها بسیار مشکل است. این مسئله عملاً در آزمون‌های کارایی و تجارب شخصی طراحان نرم‌افزار اثبات شده است. IUI با بررسی عواملی که باعث این امر می‌شوند پایه ریزی شده است. طراحان نرم‌افزارهایی که از این رابط کاربر استفاده می‌کنند باید همانند طراحان دیگر خود را به جای ساده‌ترین کاربران قرار دهند و محیطی را ایجاد کنند که از هر نظر روشن و قابل فهم باشد.

مهمترین دلایلی که باعث سردرگمی کاربران و مشکل شدن استفاده از نرم افزار می شوند عبارتند از :

- کاربران قادر نیستند که شمایی از ساختار کلی نرم افزار را در ذهن خود شبیه سازی کنند. رابط کاربر بیشتر نرم افزارها با فرض اینکه کاربر نمایی کلی از آنچه که برنامه نویس با دقت فراوان آن را ایجاد کرده است، در ذهن خود خواهد ساخت، طراحی شده است. متأسفانه باید گفت که بیشتر کاربران حتی نمی توانند تصویری سطحی از نرم افزار را برای انجام خواسته های خود ایجاد کنند. این کاربران، کندذهن و یا نادان نیستند بلکه به علت مشغله های کاری و حجم زیاد اطلاعات متفاوت نمی توانند این کار را انجام دهند. آنها زمان، نیرو و حتی علاقه ای به بررسی نرم افزار و فراگیری ساختار آن ندارند.

- حتی بسیاری از کاربران قدیمی هم قادر نیستند تا بر رویه های پرکاربرد مسلط شوند. طراحان نرم افزار بر این مسئله که کاربران در ابتدا ممکن است برای کار کردن با نرم افزار با مشکل مواجه باشند واقفند ولی انتظار دارند که این مشکل با گذشت زمان از بین برود. برخلاف این تصور و بر اساس داده های کاربردی در بیشتر موارد این امر اتفاق نمی افتد. در یک پژوهش، محققان یک سری دستگاه ویدئوی مجهز و خودکار را در اختیار کاربران خانگی قرار دادند و مشاهده کردند که کاربران لزوماً توجهی به مراحل که برای استفاده از دستگاه جدید از آن عبور می کنند ندارند و طرز کار دستگاه را به خاطر نمی سپارند. بیشتر کاربران در دومین استفاده نیز همچنان به دنبال پیدا کردن روش کارکرد وسیله بودند.

- کاربران مجبورند زمان زیادی را صرف پیدا کردن راه درست انجام خواسته ی خود بکنند. بیشتر نرم افزارها برای کاربرانی طراحی شده اند که با ساختار نرم افزار و طرز کار آن آشنایی دارند. کار کردن با بخش های مختلف یک نرم افزار برای اکثر کاربران مثل حل کردن یک جدول مشکل و ناخواسته به نظر می رسد. ممکن است کاربران تصور کنند که حل کردن این جدول ها بخش جدایی ناپذیری از کامپیوتر است ولی به طور حتم چنانچه با چنین جدول هایی روبرو نباشند راضی تر خواهند بود.

بهترین راه حل برای مشکلات فوق پیدا کردن یک استراتژی کلی برای واضح و بدیهی کردن بخش های گوناگون یک نرم افزار است. کاربران باید بتوانند به آسانی هر بخش از نرم افزار را که مایل بودند بیابند و هر عمل دلخواه خود را در حداقل زمان ممکن عملی سازند.

رابط کاربر قیاسی

بیشتر عناصر نرم افزارهای امروزی نیازمند بررسی و تعیین کارایی و نوع عملکرد توسط کاربر هستند. برای مثال به تصویر شماره ی ۱ توجه کنید.



تصویر شماره‌ی ۱ - چه کاری باید انجام دهم؟
چگونه متوجه شوم که باید این پنجره را ببندم؟

کاربران باتجربه‌ی کامپیوتر و از آن جمله طراحان نرم‌افزار به زودی در خواهند یافت که این پنجره به آنها کمک خواهد کرد تا لیستی از چیزها را مدیریت کنند. آنها متوجه می‌شوند که دکمه‌های موجود در زیر لیست برای اضافه کردن، حذف کردن و گرفتن اطلاعاتی درباره‌ی عناوین موجود در لیست هستند. توجه کنید که هیچ یک از این دانسته‌ها در خود پنجره ذکر نشده است.

و حالا خود را به جای یک کاربر ساده‌ی نرم‌افزار فرض کنید. بسیاری از کاربران در هنگام برخورد با چنین پنجره‌ای از خود خواهند پرسید «با این پنجره چه باید بکنم؟». در هنگام ظاهر شدن پنجره، کاربر باید توقف کند و فکر کند که با این اشیا چه کاری باید انجام دهد. در ابتدا کاربر باید این حقیقت را متوجه شود که جعبه‌ی بزرگ و سفیدرنگ یک لیست خالی از چیزهاست که وی باید آن را پر کند. عبارت «**Things**» در بالای لیست، اطلاعاتی مبهم در اختیار کاربر قرار می‌دهد. بعضی از کاربران تلاش خواهند کرد که در لیست، متنی را تایپ کنند زیرا شباهت زیادی با یک جعبه‌ی متن دارد.

در قدم بعدی کاربر باید متوجه شود که دکمه‌های زیر لیست برای تحت تاثیر قرار دادن محتویات لیست تعبیه شده‌اند. بعضی از دکمه‌ها به طور پیش فرض غیر قابل دسترسی هستند که خود یک عامل سردرگم کننده محسوب می‌شود. کاربر باید زمانی را صرف کند تا فقط متوجه شود که هدف از نمایش این پنجره چیست و چگونه باید با آن ارتباط برقرار کرد.

کاربر حتی ممکن است درگیر مسائل پیچیده‌تری شود مانند: «چند شیء را باید در این لیست اضافه کنم؟»، «آیا اشیا را باید طبق ترتیبی خاص به لیست اضافه کنم؟»، «چرا در اولین گام این پنجره به نمایش درآمده؟»، «هدف این پنجره چیست؟» و

چرا یک پنجره نباید عنوانی شفاف که نمایانگر هدف آن است، داشته باشد؟ هنگامی که طراحان پنجره‌ای را ایجاد می‌کنند به ندرت به این مسئله که هدف اصلی از ساخت آن چیست، می‌اندیشند. و عمدتاً هر صفحه بخشی از یک فرآیند بزرگتر و پیچیده‌تر است که کاربر باید بر آن مسلط شود.

تحقیقات نشان داده‌اند که حتی فرآیندهای ساده هم باعث سردرگمی کاربران می‌شوند. آنها برای اینکه متوجه شوند که هدف از نوشته شدن این نرم‌افزار چیست، برای انجام عمل دلخواه خود به چه بخشی باید مراجعه کنند و پس از پیدا کردن بخش مورد نظر چگونه با آن کار کنند، با مشکل مواجه هستند.

ساده کردن نرم افزار با ایجاد تغییرات بنیادین در آن راهی است موثر برای راضی کردن مشتریان فعلی و جذب مشتریان جدید.

یک راه حل: رابط کاربر القایی

به عنوان راهی جدید برای طراحی نرم افزار، هدف از IUI، کاهش زمانی است که باید صرف شود تا کاربر بتواند بر بخش های مختلف یک نرم افزار تسلط پیدا کند.

IUI ادامه ای است از روش طراحی صفحات اینترنتی. در اینترنت، صفحات باید تا حد امکان ساده و هدفدار باشند زیرا هر بخش از اطلاعات می بایست از طریق یک مسیر عمدتاً کم سرعت به کاربر برسد. کاربر به هر بخش از اطلاعات پاسخ می دهد و سپس سرور بخش بعدی را در اختیار وی قرار می دهد. طراحی خوب صفحات اینترنتی به معنای تمرکز بر روی یک وظیفه و قرار دادن امکان حرکت در بین صفحات است. به طور مشابه، هدایت کننده ی القایی نیز بر همین اساس پی ریزی شده است.

یک صفحه که بر اساس اصول IUI طراحی شده باشد به کاربر کمک می کند تا به این دو سوال بنیادین که در هنگام مشاهده ی آن مطرح می شود پاسخ دهد:

– هم اکنون چه کاری باید بکنم؟

– پس از به پاسخ به خواسته های این صفحه به کجا باید بروم؟

نرم افزاری که از IUI استفاده می کند با تکیه بر این قضیه ی بنیادی که: فهم صفحه ای بایک هدف ساده، صریح و مشخصاً بیان شده بسیار ساده تر از صفحه ای است که این هدف را نداشته باشد، به این پرسشها پاسخ می دهد. هر قدر صفحه ای ساده تر باشد، درک عملیات جاری و کارهای آتی برای کاربر آسان تر خواهد بود.

بر اساس این قضیه می توان مراحل را که باید برای ایجاد یک برنامه ی مبتنی بر IUI پشت سر گذاشت به چهار بخش زیر تقسیم بندی کرد:

۱- در هر صفحه بر روی یک هدف خاص تمرکز کنید.

۲- هر بخش را کاملاً شرح دهید.

۳- محتویات صفحه را بر اساس هدف آن پی ریزی کنید.

۴- گزینه هایی را برای انجام کارهای ثانویه و جانبی در اختیار کاربر قرار دهید.

این مقاله در کنار شرح اصول IUI، مثالهایی عملی از کاربرد آن را در MS Money 2000 و دیگر نرم افزارها به شما نشان خواهد داد. شما باید به این مثالها به عنوان نمایشی از این نوع رابط کاربر توجه کنید نه مدلی برای پیاده سازی بر روی دیگر نرم افزارها.

علاوه بر چهار نکته‌ی ذکر شده، باید با کمک گرفتن از راهبردهای زیر رابط کاربر نرم‌افزار خود را قدرتمند کنید:

- در تمام بخش‌های برنامه از قالبی ثابت و پایدار استفاده کنید.
- صفحاتی را برای شروع کارها در نظر بگیرید.
- روش کارکردن با اشیا و ارتباط با فرآیندها را به روشنی شرح دهید.
- راهی آسان برای به پایان رساندن هر وظیفه و شروع وظیفه‌ی بعدی طراحی کنید.
- به روشنی توضیح دهید که قدم بعدی چیست.

این چهار قدم و پنج راهبرد اصول طراحی یک رابط کاربر القایی را تشکیل می‌دهند.

فرآیندها

بسیاری از وظایف نیازمند حرکت کاربر مابین چندین صفحه هستند. کاربری که قصد انجام هدفی را دارد ممکن است با مراجعه به یک صفحه‌ی جانبی از مراحل اصلی کامل کردن یک فرآیند خارج شود. پس از آنکه عملیات ثانویه به پایان رسید باید راهی روشن و ساده برای بازگشت به صفحه‌ی ارجاع دهنده وجود داشته باشد. به نظر می‌رسد که کاربران برای کار کردن با روش‌های قدیمی ارجاع (مانند دکمه‌های **Next** و **Back**) مشکل داشته باشند.

برای ایجاد این توانایی، **UI مدلی** به نام **فرآیند** را معرفی می‌کند. یک فرآیند مثل یک سبب زیرروال (**Subroutine**) عمل می‌کند. کاربران قادرند تا یک فرآیند را آغاز کنند، مابین صفحات آن حرکت کنند و در پایان با فشردن دکمه‌ی «پایان» به عملیات خاتمه دهند تا به صفحه‌ای که فرآیند را شروع کرده‌اند، بازگردند.

مراحل ایجاد یک رابط کاربر القایی

این بخش به تفصیل درباره‌ی چهار قدم اساسی در ساخت **UI** خواهد پرداخت.

گام اول: در هر صفحه بر روی یک هدف خاص تمرکز کنید.

اولین گام در ساخت **UI** انتخاب یک وظیفه‌ی مشخص، تعیین اهداف و تقسیم آن‌ها به صفحات مجزا است. برای هر هدف معین باید یک صفحه ایجاد شود و در آن صفحه فقط به همان هدف که هدف اولیه‌ی آن صفحه نامیده می‌شود پرداخته شود.

این نکته ساده به نظر می‌رسد ولی تعداد اندکی از برنامه‌ها به آن پایبند هستند. بیشتر نرم‌افزارها علاوه بر هدف اولیه به چندین عملکرد مرتبط دیگر نیز می‌پردازند و کاربر را وادار می‌کنند تا خود به پیدا کردن راه حل مشکلاتش بپردازد.

مثال

در این قسمت به بررسی دو نسخه‌ی متفاوت از MS Money خواهیم پرداخت. در مثالهای ذکر شده به بخش‌های مشابهی که وظیفه‌ی مدیریت حسابهای مالی را بر عهده دارند اشاره شده است.

درباره‌ی Money 2000 و رابط کاربر القایی

رابط کاربر القایی در حین ساخت این نرم‌افزار به وجود آمد.

Money 2000 یک نرم‌افزار به شکل صفحات اینترنت است. این نرم‌افزار یک سایت نیست ولی با یک سایت اینترنتی شباهت‌های زیادی دارد. این برنامه برای ارتباط با کاربر از صفحات کامل که همگی در یک قاب کلی مشترکند استفاده می‌کند و در عین حال، امکان سیاحت آسان در بخش‌های مختلف وجود دارد.

اگر چه IUI برای اولین بار در رابط کاربر Money 2000 که به صفحات اینترنتی شباهت دارد مورد استفاده قرار گرفته ولی می‌توان با استفاده از اشیائی که در رابط‌های سنتی استفاده می‌شده نیز، رابط کاربر القایی را بر روی نرم‌افزار خود پیاده‌سازی کرد.

در Money 99 کاربران اغلب در یک صفحه می‌توانستند چندین کار مختلف را انجام دهند. برای مثال تصویر شماره‌ی ۲ نشان‌دهنده‌ی صفحه‌ای به نام مدیر حساب‌ها است که تمامی اعمال مربوط به حساب‌ها را بر روی آنها انجام می‌دهد.

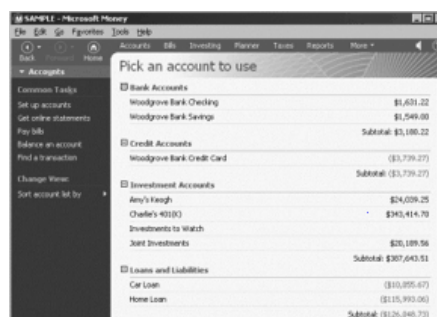


تصویر شماره‌ی ۲ - بخش مدیریت حسابها در Money 99

تمامی عملیات مربوط به حساب‌ها در این پنجره مشاهده می‌شود

در این صفحه عمل عمومی مدیریت حساب‌ها به نمایش در آمده است. در این صفحه می‌توان اعمالی مانند ایجاد و حذف یک حساب را، که به ندرت انجام می‌شوند، به اجرا در آورد. هیچ یک از اعمالی که در این صفحه می‌توان انجام داد به روشنی در عنوان آن ذکر نشده است و تشخیص این اعمال بر عهده‌ی کاربر می‌باشد.

Money 2000 که از IUI استفاده می‌کند، تقریباً اعمال مشابهی را بر روی حساب‌ها به انجام می‌رساند ولی در دو صفحه‌ی کاملاً مجزا. تصویر شماره‌ی ۳، صفحه‌ی اول را که بر روی انتخاب حساب متمرکز شده است، نمایش می‌دهد.



تصویر شماره‌ی ۳ - صفحه‌ای از Money 2000 که فقط

بر روی انتخاب حساب متمرکز شده است.

صفحه‌ی Money 2000 تقریباً همان تعداد عناصر تصویری که Money 99 دارد را در خود جای داده است ولی این صفحه فقط بر روی انتخاب حساب متمرکز شده است. در نسخه‌ی قدیمی کاربر برای باز کردن یک حساب مجبور بود دو بار کلیک کند. یک بار بر روی عنوان حساب برای انتخاب آن و بار دیگر بر روی دکمه‌ی باز کردن. ولی در نسخه‌ی 2000 این عمل در یک کلیک خلاصه شده است زیرا هدف از این صفحه باز کردن یک حساب است. البته واضح است که این کار باعث افزایش صفحات برنامه شده است ولی در عوض وضوح برنامه افزایش یافته است و هر کاربر ساده، به راحتی می‌تواند با برنامه ارتباط برقرار کند.

بعضی اوقات کاربران مایل به حذف و یا ایجاد یک حساب هستند. این اعمال در صفحه‌ی دوم که در تصویر ۴ مشاهده می‌کنید، انجام می‌شوند.



تصویر شماره‌ی ۴ - این صفحه از Money 2000 بر روی عملیاتی که می‌توان بر روی یک حساب انجام داد، متمرکز شده است.

یک وظیفه‌ی تنها چیست ؟

چگونه می‌توان فهمید که یک صفحه بر روی یک وظیفه متمرکز شده است؟ ممکن است صفحه‌ای برای چندین عمل طراحی شده باشد ولی در توضیحات فقط به یک وظیفه‌ی آن اشاره شده باشد. در اینجا به یک قانون کلی اشاره می‌کنیم: یک صفحه، تنها زمانی بر روی یک موضوع متمرکز شده است که طراح بتواند آن را با یک عبارت کوتاه، با معنی و آشنا با ذهن کاربر توضیح دهد.

اگر در هنگام انتخاب عنوان برای یک صفحه با مشکل برخورد کردید، ممکن است که در آن صفحه بر روی یک موضوع خاص تکیه نکرده باشید.

گام دوم: هر بخش را کاملاً شرح دهید.

هر صفحه باید با یک عنوان روشن و مختصر شرح داده شود که می‌تواند یک دستورالعمل مستقیم «حساب مورد نظر خود را انتخاب کنید» و یا سؤالی که انتظار دارید کاربر به آن پاسخ دهد «تراز مالی کدام حساب را می‌خواهید؟»، باشد.

تعیین عنوان هر صفحه باید در حین ساخت آن انجام گیرد و چنانچه این عمل به انتهای کار موکول شود، ممکن است عبارتی مناسب، کوتاه و معنی‌دار قابل انتخاب نباشد و در نتیجه‌ی تعیین عنوان نامناسب، وضوح برنامه کاهش یابد.

در مدل IUI عنوان هر صفحه در اولین مرحله تعیین می‌گردد و سپس شکل ظاهری صفحه و توانایی‌های آن

براساس عنوان مشخص شده، طراحی می‌شوند.

هنگامی که عنوانی برای یک صفحه انتخاب شد نباید تصور شود که این عنوان غیر قابل تغییر است. بلکه ممکن است بنابر نظر گروه و طراح عنوانی روشن‌تر برای آن صفحه تعیین شود. ولی همیشه عنوان ابتدایی باید کامل و دقیق باشد زیرا عنوان صفحه، شکل طراحی را تحت تأثیر خود قرار می‌دهد.

راهنمای انتخاب عنوان

این قسمت به شرح راهکارهایی ساده برای انتخاب بهترین عنوان‌ها می‌پردازد. برای این کار کافی است دوستی را تصور کنید که از شما می‌پرسد «این صفحه برای چه کاربردی ساخته شده است؟». و حالا تلاش کنید که پاسخی کوتاه، و کامل به این پرسش دهید. مانند «اینجا بخشی است که شما می‌توانید...». کلماتی که این جمله را تکمیل می‌کنند می‌توانند عنوانی مناسب برای صفحه‌ی شما باشند.

در انتخاب عنوان باید تلاش کرد که از ترکیبات عطفی («یا»، «و») استفاده نشود. زیرا استفاده از این لغات به این معنی هستند که صفحه‌ی شما بر روی یک عنوان متمرکز نشده است. در انتخاب عنوان، بهتر است از جملات به جای کلمات استفاده کنید تا به برنامه‌ی خود وضوح بیشتری دهید. مانند «یک حساب را انتخاب نمایید» به جای «انتخاب حساب».

حتی پس از انتخاب یک عنوان مناسب، ممکن است صفحه همچنان نامفهوم باقی مانده باشد. برای حل این مشکل از توضیحاتی در زیر عنوان یا در پایین صفحه که عملکرد دقیق صفحه را نشان می‌دهند استفاده نمایید.

عنوان صفحات را برجسته کنید.

تحقیقات به انجام رسیده حاکی از این مطلب هستند که بیشتر کاربران تمایلی به خواندن توضیحات ندارند بنابراین یک عنوان مناسب و کامل، باید در محلی که کاملاً جلب توجه کند قرار بگیرد و کاربر با مشاهده‌ی صفحه، ناخودآگاه به عنوان صفحه هدایت شود.

گام سوم: محتویات صفحه را بر اساس هدف آن پی‌ریزی کنید.

اشیائی که در هر صفحه قرار می‌گیرند باید با دقت فراوان و مناسب با عنوان آن صفحه انتخاب شوند. چنانچه اشیائی که از آنها استفاده می‌شود با هدف صفحه همخوانی کافی نداشته باشند، کارایی برنامه کاهش خواهد یافت. بهتر است از اشیاء استاندارد محیط سیستم عامل استفاده کنید چون بیشتر کاربران با این اشیاء و طرز کار آنها آشنایی قبلی دارند و نیازی به صرف زمان اضافی برای فراگیری روش کارکردشان نخواهند داشت.

راهی روشن برای خروج از صفحه تعبیه کنید.

در صفحاتی که پی‌پی به نمایش در می‌آیند زمان خروج مشخص است ولی در صفحاتی که به نمایش یا دریافت

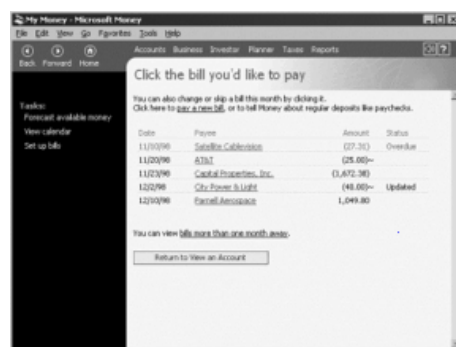
اطلاعات اختصاص دارند، ممکن است تعیین زمان و روش خروج برای کاربر مشکل باشد. در چنین صفحاتی روشی مشخص مانند قرار دادن دکمه‌ی «بعدی» یا «پایان» در مکانی قابل رؤیت، برای خروج تعبیه کنید.

گام چهارم: گزینه‌هایی را برای انجام کارهای ثانویه و جانبی در اختیار کاربر قرار دهید.
همیشه اتصالاتی برای اعمال ثانویه در اختیار کاربر قرار دهید. برای مثال در صفحه‌ای که هدف آن نوشتن یک نامه است گزینه‌های ثانویه می‌توانند «چاپ پاکت نامه» و «جستجوی آدرس‌های پستی» باشند. این اتصالات می‌توانند راهنمایی برای کاربرانی که مسیر را اشتباه انتخاب کرده‌اند و ابزاری برای کارایی بیشتر برنامه باشند.

توجه کنید که ممکن است در یک صفحه چند و در صفحه‌ای دیگر هیچ گزینه‌ای برای اعمال ثانویه نباشد. در صفحاتی که چندین عمل ثانویه ممکن است انجام شود، مهمترین گزینه‌ها را در ابتدا قرار دهید و از افزایش بیش از حد این انتخاب‌ها پرهیز کنید زیرا باعث به هم ریختگی رابط کاربر شما خواهند شد. چنانچه تعداد این انتخاب‌ها زیاد است و تمامی آنها مورد نیاز هستند بهتر است چند انتخاب که از همه مهم‌تر هستند را در ابتدا ذکر کنید و بقیه را در صفحه‌ای مجزا قرار دهید. سپس با قرار دادن اتصالی به آن صفحه، کاربر را به آنجا هدایت کنید.

محل قرارگیری گزینه‌های جانبی

این انتخاب‌ها باید در محلی ثابت و قابل رؤیت قرار بگیرند تا کاربر در هر صفحه در صورت نیاز به راحتی آنها را بیابد. اتصالات نباید با اشیاء اصلی برنامه مخلوط شوند و باید در محلی جدا و معین که فقط برای همین امر در نظر گرفته شده است، قرار بگیرند. در تصویر شماره‌ی ۵ می‌توانید محل قرارگیری این اتصالات را در Money 2000 مشاهده کنید.



تصویر شماره‌ی ۵

محل قرارگیری اتصالاتی که در سمت چپ پنجره

اگر قصد قرار دادن انتخاب‌های ثانویه را در کنار پنجره دارید، این انتخاب‌ها نباید قابل پیمایش به بالا و پایین باشند و همچنین نباید همراه اطلاعات اصلی حرکت کنند.

راهکارهای تکمیلی

این قسمت به شرح بیشتر پنج راهکار ذکر شده در بخش‌های گذشته می‌پردازد. با بهره‌گیری از چهار قدم ذکر شده و پنج راهکار زیر، شما آمادگی ساخت یک نرم‌افزار بر اساس IUI را خواهید داشت.

در تمام بخش‌های برنامه از قالبی ثابت و پایدار استفاده کنید.

هنگامی که قصد استفاده از **IUI** را دارید ابتدا باید قالبی ثابت طراحی کنید تا در تمامی صفحات از آن استفاده نمایید. هدایت‌کننده‌ی قالبی مدل خاصی را پیشنهاد نمی‌کند. برای ایجاد یک رابط‌القایی راه‌های فراوانی وجود دارد. ممکن است شما از یک قالب ثابت برای تمام صفحات استفاده کنید و امکان دارد که چند قالب متفاوت برای اهداف گوناگون طرح‌ریزی نمایید.

یک قالب خوب به یک کاربر ساده و جدید کمک می‌کند تا به آسانی با برنامه ارتباط برقرار کند. اگر محل قرارگیری اشیا در صفحه همیشه ثابت باشد، مشتری نیازی به جستجو برای پیدا کردن بخش مورد نظر خود در هر صفحه نخواهد داشت.

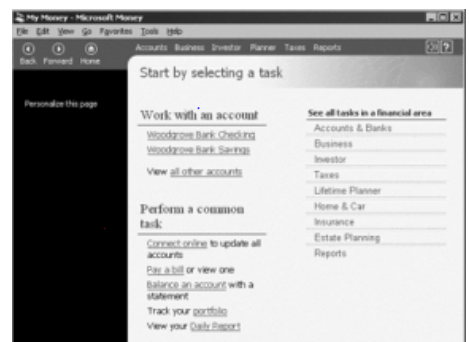
صفحاتی را برای شروع کارها در نظر بگیرید.

نرم‌افزارهایی که از **IUI** استفاده می‌کنند، صفحاتی برای تعیین مقصد کاربر در اختیار وی قرار می‌دهند. در این صفحات از کاربر پرسش می‌شود که اکنون قصد چه کاری را دارد و چندین پاسخ احتمالی در اختیار وی قرار می‌گیرد تا با انتخاب هریک به بخش مناسب هدایت شود. به این گونه صفحات، صفحات فعالیت گفته می‌شود. نمایش یک صفحه فعالیت در ابتدای برنامه باعث جلوگیری از سردرگمی کاربر خواهد شد.

صفحه فعالیت‌تی که در ابتدای برنامه به نمایش در می‌آید، باید تمام انتخاب‌هایی را کاربر ممکن است به آن دلیل برنامه را اجرا کرده، در بر داشته باشد. برای انجام این کار باید پرکاربردترین اعمالی را که در برنامه می‌توان انجام داد گروه‌بندی کرد و در ابتدای کار در اختیار کاربر قرار داد.

در تصویر شماره‌ی ۶ می‌توانید یک نمونه از صفحه فعالیت را مشاهده کنید.

تصویر شماره‌ی ۶ - بخش ابتدایی **Money 2000** یک صفحه فعالیت است
در این بخش اتصالاتی برای چند وظیفه‌ی پرکاربرد و چند گروه از اعمال در صفحات دیگر تعبیه شده است



روش کارکردن با اشیا و ارتباط با فرآیندها را به روشنی شرح دهید.

بهترین راه برای دنبال کردن این راهبرد، استفاده از یک عنوان مناسب و محدود کردن حوزه‌ی صفحه به هدف اولیه

است. اگر از یک عنوان مناسب برای صفحه استفاده کنید و هدف اصلی صفحه را به روشنی تعیین نمایید، انتخاب اشیاء مناسب کاری مشکل نخواهد بود.

راهی آسان برای به پایان رساندن هر وظیفه و شروع وظیفه‌ی بعدی طراحی کنید.

آخرین سدی که در برابر کاربر قرار می‌گیرد، زمان و چگونگی خروج از صفحه‌ی فعلی است. معمولاً کاربر با کلیک بر روی عنوانی که وی را به صفحه‌ی بعدی هدایت می‌کند و یا اجرای دستوری که باعث انتقال فرمان به بخشی دیگر می‌شود، از صفحه‌ی فعلی خارج می‌شود. این اتّصال‌ها می‌توانند در داخل متن برنامه، در کنار صفحه یا بر روی نوارهای ابزار قرار بگیرند. همچنین ممکن است کاربر با بستن فایل باز یا بستن نرم‌افزار از این صفحه خارج شود.

شروع و پایان فرآیندها

این بخش به شرح فرآیندها اختصاص دارد. یک فرآیند شیوه‌ای برای ارتباط با کاربر در بخش‌هایی است که شامل بیش از یک صفحه هستند.

کاربری را تصوّر کنید که بر روی یک اتّصال در متن برنامه یا لیست اعمال جانبی کلیک کند و به صفحه‌ای دیگر هدایت شود. آن صفحه ممکن است بخش ابتدایی از یک سری صفحات باشد که وظیفه‌ای را به انجام خواهند رساند. پس از پایان عملیات کاربر قصد بازگشت به صفحه‌ی اولیه را دارد. برای این منظور دو راه وجود دارد. راه اول فشردن کلید «Back» تا جایی است که به صفحه‌ی مورد نظر برسد و راه دیگر خروج از این بخش، مراجعه به صفحه‌ی ابتدایی برنامه و باز کردن مجدد صفحه‌ی قبلی می‌باشد. هیچ یک از روش‌های فوق منطقی به نظر نمی‌رسند. کاربران انتظار دارند که در صفحه‌ی پایانی هر فرآیند راهی برای انتقال کنترل به صفحه‌ی ارجاع دهنده وجود داشته باشد.

در مدل IUI این امکان وجود دارد که کاربر پس از به پایان رساندن هر فرآیند دقیقاً به بخش ارجاع دهنده برگشت کند.

نام فرآیند

نام هر فرآیند باید به دقت انتخاب شود و در تمام صفحاتی که فرآیند شامل می‌شود، ذکر شوند. این نام باید با تأکیدی کمتر از عنوان اصلی صفحه وجود داشته باشد تا کاربر در هر زمان بتواند تشخیص دهد که این صفحه بخشی از یک فرآیند کلی است و به تنهایی عملکردی ندارد.

پیاده‌سازی فرآیندها

یک فرآیند می‌تواند از چندین صفحه‌ی متفاوت به اجرا درآید. انتقال کنترل برنامه به صفحه‌ی ارجاع دهنده نمی‌تواند از راه کد نویسی در اتّصال پایانی به انجام برسد. زیرا ممکن است در طی گسترش برنامه صفحات

جدیدی ایجاد شوند و کنترل این صفحات در یک پروژه‌ی بزرگ عملاً کاری است ناشدنی.

راه حلّ مشکل ایجاد یک پشته (Stack) جداگانه برای صفحات باز شده است. این پشته باید در سراسر برنامه قابل دسترسی باشد و با باز شدن هر پنجره‌ی جدید، نام آن در ابتدای پشته قرار بگیرد. با پیاده‌سازی این روش، بازگشت به صفحه‌ی قبلی بسیار آسان خواهد بود. توجه کنید که این پشته نباید با پشته‌ای که در فرآیندها مورد استفاده قرار می‌گیرد ترکیب شود و هر فرآیند علاوه بر دسترسی به پشته‌ی سراسری برنامه باید پشته‌ای مجزاً برای صفحاتی که در اختیار دارد، داشته باشد.

دکمه‌ی پایان

هر فرآیند در بخش‌های پایینی خود شامل دکمه‌های «قبلی»، «بعدی» و «پایان» و یا دکمه‌های مشابه می‌باشد. این دکمه‌ها وظیفه‌ی هدایت کاربر در بین صفحات را به عهده دارند. چنانچه یک فرآیند از چندین بخش صدا شود، دکمه‌ی پایان می‌تواند شامل نام صفحه‌ی ارجاع دهنده باشد. مانند: «بازگشت به بخش مالیات‌ها».

گام بعدی را به روشنی توضیح دهید.

اندکی از برنامه‌ها تمام بخش‌های یک وظیفه را در یک صفحه می‌گنجانند و بنابراین کاربر مجبور است با حرکت در بین چند صفحه یک وظیفه را به پایان برساند. حرکت در بین صفحات متعدّد می‌تواند باعث سردرگمی کاربر شود. چنانچه مراحل بعدی که باید به انجام برسند، برای کاربر مشخص باشند، این سردرگمی به حداقل خواهد رسید و ارتباط با برنامه بسیار ساده‌تر خواهد بود.

هدایت کاربر

این بخش به شرح چند راهکار برای قرار دادن توضیحاتی جهت راهنمایی کاربران در یک برنامه می‌پردازد. هدایت اولیه به تمام متن‌هایی که در برنامه مشاهده می‌شوند مرتبط می‌شود. یک برنامه‌ی مبتنی بر **GUI** باید طوری طراحی شود که کاربر بتواند هر عملی را که نیاز داشت به راحتی پیدا کند و به آسانی متوجه شود که هر اتصال موجود در این صفحه چه وظیفه‌ای را برای وی به انجام خواهد رساند. در تصویر شماره‌ی ۷ می‌توانید نمونه‌ای از یک صفحه‌ی روشن و واضح را مشاهده کنید.

تصویر شماره‌ی ۷ - عباراتی برای توضیح هر وظیفه می‌تواند

ارتباط کاربر و نرم‌افزار را آسان‌تر نماید.



هدایت ثانویه به تمامی متن‌هایی گفته می‌شود که بدون خواست کاربر قابل مشاهده نیستند. مانند متن‌هایی که در هنگام انتقال نشانگر ماوس بر روی اشیاء به نمایش در می‌آید. این متن‌ها به طور مستقیم در هدایت کاربر

نقشی ندارند ولی بعضی اوقات می‌توانند باعث راهنمایی وی شوند.

هدایت اولیه

هدایت اولیه شامل تمام و یا بخشی از مؤلفه‌های زیر می‌باشد.

۱- عنوان صفحه

مثال: تصویر خود را عوض کنید.

عنوان صفحه، اولین و مهم‌ترین بخشی است که در صفحه ظاهر می‌شود. هدف از قرار دادن عنوان، راهنمایی کاربر در مورد کاری است که در این صفحه می‌تواند انجام دهد و یا باید انجام دهد. عنوان صفحه نباید شامل راه کامل کردن وظیفه باشد و فقط باید به ذکر وظیفه اکتفا کند.

۲- زیرنویس صفحه

مثال: همچنین می‌توانید یک تصویر را از روی اینترنت دانلود کنید.

با وجود تلاش فراوان ممکن است عنوان صفحه نتواند دربرگیرنده‌ی یک وظیفه‌ی پیچیده باشد. یک زیرنویس خوب باید شامل توضیحاتی که تکمیل‌کننده‌ی عنوان صفحه است، باشد و همان طور که در مورد عنوان ذکر شد، زیرنویس نیز نباید به شرح روش کار با این صفحه بپردازد.

۳- وظایف

مثال: محافظ صفحه‌ی نمایش خود را عوض کنید.

وظایف می‌توانند از متن و یا تصاویر تشکیل شده باشند. وظایف برای به اجرا درآمدن نیازمند عکس‌العمل کاربر هستند. وظایفی که به شکل متنی به نمایش درمی‌آیند باید بر اساس یک فعل نوشته شوند و از جملاتی کوتاه و گویا استفاده نمایند.

۴- برچسب دکمه‌ها

مثال: ایجاد رمز عبور

به طور کلی سه نوع دکمه وجود دارد:

- الغا

- پایان

- اجرا

دکمه‌های الغا و پایان می‌توانند به سادگی دارای برچسب هم‌نام خود باشند ولی دکمه‌هایی که برای اجرای یک فرمان به کار برده می‌شوند باید از یک برچسب گویا -مانند «ایجاد رمز عبور» به جای «ادامه»- استفاده کنند.

۵- برچسب دیگر اشیا

مثال: رمز عبور خود را تایپ نمایید.

برچسب اشياء دیگر باید مختصر، گویا و دقیقاً بیانگر کاری باشد که کاربر باید انجام دهد تا یک وظیفه به پایان برسد.

۶- اتصال برای وظایف مرتبط

مثال: وظایف مرتبط:

یک حساب دیگر را تغییر بده.

این اتصالات باید روشن باشند و بر اساس وظایف نوشته شوند.

۷- اتصال برای «رجوع کنید به...»

مثال: رجوع کنید به...

تغییر شکل پنجره‌ها

این اتصالات کاربر را به وظایف ثانویه هدایت می‌کنند و اتصالات آنها باید همانند اتصالات به دیگر وظایف ایجاد شود.

هدایت ثانویه

هدایت ثانویه می‌تواند شامل مؤلفه‌های زیر باشد

۱- InfoTip‌ها

InfoTip‌ها می‌توانند برای دادن اطلاعات اضافی و کمکی در مورد اشیا و کارها مورد استفاده قرار بگیرند. InfoTip‌ها پنجره‌های کوچکی هستند که در هنگام قرارگیری نشانگر ماوس بر روی اشیا به نمایش در می‌آیند و توضیحاتی برای تکمیل درک کاربر از محیط ارائه می‌دهند. شما باید برای تمامی اشیائی که قابلیت کلیک شدن دارند، یک توضیح کمکی در InfoTip قرار بدهید. مانند: «صفحه‌ای که در آن می‌توانید تصویری برای خود انتخاب کنید، به نمایش در خواهد آمد».

۲- عنوان‌های کمکی «فراگیری...»

مثال: فراگیری...

دانلود کردن یک فایل

اتصالات‌های فراگیری، صفحات راهنمای برنامه را که شامل خلاصه‌ی توانایی‌ها، اطلاعات ادراکی، اطلاعات پشتیبانی و اطلاعات جاری می‌شوند، نمایش می‌دهند. برای جلوگیری از به هم ریختگی صفحه، تعداد این عناوین نباید زیاده‌تر از حد معمول باشد.

مقایسه با سایت‌های اینترنتی

بسیاری از سایت‌هایی که با مهارت و دانش طراحی شده‌اند از اصولی مشابه اصول **IUI** استفاده می‌کنند. این شباهت می‌تواند به دلیل محدودیت‌هایی باشد که در اینترنت مطرح هستند. مانند سرعت پایین، عدم دسترسی به اشیائی که در تولید نرم‌افزار موجود هستند و نیاز بیشتر به وضوح صفحه.

طراحان نرم‌افزارهای سنتی امکان استفاده از اشیاء زیادی را داشتند و همین امر باعث راحت تر شدن فرآیند تولید نرم‌افزار می‌شد و در عوض ممکن بود باعث پیچیدگی کاربرد نرم‌افزار شود. در طراحی نرم‌افزارهای جدید که از **IUI** استفاده می‌کنند برنامه‌نویس باید در استفاده از اشیاء احتیاط بیشتری اعمال کند و نباید ارزش وضوح و سادگی برنامه را فراموش نماید.

Microsoft Developer Network

منبع :

امیر مسعود ایرانی (AMIB)

ترجمه با تلخیص :

AMIBTemp@Hotmail.com

یکشنبه ۲۹ آذر ۸۳