

Usabilidade

Usabilidade	1
Usabilidade, accesibilidade e experiencia de usuario (UX)	1
Usabilidade: características e atributos	2
Medida de usabilidade de aplicacións: tipos de métricas	3
Pautas de deseño de interfaces	3
Deseño da estrutura das interfaces	4
Aspecto das interfaces	4
Elementos interactivos	5
Outras pautas de deseño	5
Presentación dos datos	5
Secuencia de control da aplicación	5
Aseguramento da información	6
Aplicacións multimedia	6
Java Swing - Look And Feel	6
Look and Feels dispoñibles	7
Exercicios	7
Referencias	9

Usabilidade

O desenvolvemento de aplicacións multiplataforma hoxe en día non pode ser entendido sen adicar unha porcentaxe bastante importante do tempo a planificar, analizar, deseñar, implementar e probar as súas **interfaces**, xa que son o medio fundamental polo cal as persoas usuarias poden comunicarse coa aplicación e realizar as operacións para as que foi deseñada.

Á hora de deseñar unha interface hai que ter en conta múltiples aspectos: **a interacción persoa-computador, a usabilidade e a accesibilidade**. Outro aspecto moi importante é a **experiencia de usuario** (UX), que é un concepto moi amplo, aínda que se pode resumir como a percepción que recibe unha persoa ao usar un produto, que pode ser positiva ou negativa. A percepción depende de moitos factores: cores usadas, fontes tipográficas, disposición dos elementos, etc.

Usabilidade, accesibilidade e experiencia de usuario (UX)

Estes tres conceptos son de grande relevancia no deseño dunha boa interface.

A **usabilidade** refírese á facilidade de uso dunha aplicación interactiva.

Segundo a Organización Internacional para a Estandarización (ISO):

- A usabilidade refírese á capacidade dun software de ser comprendido, aprendido, usado e ser atractivo para as persoas usuarias, en condicións específicas de uso (ISO/IEC 9126).
- Usabilidade é a eficacia, eficiencia e satisfacción coa que un produto permite alcanzar obxectivos específicos a persoas usuarias específicas nun contexto de uso específico (ISO/IEC 9241).

Segundo **Jakob Nielsen**, a usabilidade é o atributo de calidade que mide o fácil que é de usar unha interface.

Unha interface usable intenta buscar que as persoas usuarias non teñan que pensar como está organizada a información. Isto incluso leva a minimizar o número de clics para completar unha tarefa.

Este concepto ten grande relevancia, xa que canto máis fácil sexa de usar unha aplicación, máis posibilidades hai de realizar de forma eficiente o traballo.

Tamén a usabilidade inflúe no éxito dunha aplicación. É dicir, non só haberá que crear software correcto (que cumpra cos requirimentos pedidos) e con unha fiabilidade boa (os resultados non poden dar lugar a dúbidas), senón que a usabilidade do software deberá ser satisfactoria.

A **accesibilidade** trata o deseño de interfaces para que sexan fáciles de percibir, operativas e comprensibles para persoas cun abanico amplo de habilidades ou con circunstancias, contornas e condicións diferentes. É dicir, a accesibilidade pode ser entendida como a usabilidade para todas as persoas.

Actualmente, no deseño da interface utilízanse metodoloxías que orientan o produto ás persoas usuarias finais, onde cada decisión tomada se basea nas necesidades, obxectivos, expectativas, motivacións e capacidades das persoas. É dicir, hoxe en día o deseño das interfaces tamén ten en conta aspectos relativos ás emocións, sentimentos e á percepción de confianza no produto ou marca corporativa da web.

A **experiencia de usuario** (*User experience UX*), é unha filosofía de deseño que ten por obxectivo a creación de interfaces que resolvan necesidades concretas de persoas usuarias finais, conseguindo así unha maior satisfacción das mesmas, así como tamén unha mellor experiencia de uso co mínimo esforzo.

O deseño baseado na experiencia do usuario ten en conta diferentes ramas como son a psicoloxía, a socioloxía, a etnoloxía, a antropoloxía, a ciencia cognitiva e a estatística.

Exercicio de apreciación: para cada unha das páxinas web seguintes, avalía e comenta a experiencia de usuario e a usabilidade:

- [Google](#)
- [Yahoo](#)
- [La peor página web del mundo](#)
- [Puertas y vallas](#)

Usabilidade: características e atributos

As dimensións da usabilidade son unha serie de características e atributos que permiten valorar o grao de cumprimento dos obxectivos establecidos:

- **eficiencia:** unha aplicación eficiente aportará máis produtividade ás persoas usuarias permitindo un uso máis rápido da mesma.
- **eficacia / efectividade:** capacidade de conseguir un obxectivo planeado.
- **satisfacción:** é o cumprimento dos obxectivos formulados inicialmente para obter uns resultados con certo grado de xúbilo para a persoa usuaria.
- **atractivo:** unha interface é atractiva para as persoas cando aceptan de bo grao as características e o uso, mostrando unha predisposición para usala.
- **facilidade de aprendizaxe:** as interfaces deben ser intelixibles e amigables para permitir o aprendizaxe intuitivo.
- **facilidade do sistema para ser recordado:** debe permitir recordar facilmente como se usa. Se as funcionalidades e iconas son difíciles de interpretar, habería que consultar continuamente os manuais.
- **tolerancia a erros:** esta característica mide o grao en que a interface evita os erros ou axuda a superalos.

Medida de usabilidade de aplicacións: tipos de métricas

Unha vez deseñada unha interface, ¿como se pode saber se cumpre os requisitos de usabilidade? ¿Como se avalía a usabilidade?

É unha pregunta difícil de responder. Hai moitos parámetros que se poden valorar. Moitos son subxectivos e difíciles de medir. Deben buscarse métricas obxectivas para valorar a usabilidade.

Poderían levarse a cabo avaliacións cualitativas, pero debe establecerse un sistema cuantitativo da usabilidade. Para definir cuantitativamente a usabilidade, definíronse algunhas métricas que intenta avaliar as aplicacións dende un punto de vista das características e atributos da usabilidade.

Para obter datos **cualitativos** hai que conseguir informacións das persoas usuarias finais a través de enquisas, preguntas con resposta aberta, opinións, etc.

Para obter datos **cuantitativos** será necesario analizar o comportamento da persoa usuaria ante a nova aplicación, recollendo informacións obxectivas como:

- tempo adicado a cada interface antes de escoller unha opción.
- tempo ou número de veces de uso da axuda ou documentación.
- frecuencia do uso de opcións.
- número de erros no uso das interfaces por persoa usuaria ou cada certo tempo.
- número de accións completadas nun tempo determinado.
- comparación do tempo usado en segundas ou terceiras visitas a unha mesma interface.
- número de suxestións ou queixas do produto.

A partir destas métricas cuantitativas pode avaliarse a usabilidade dende o punto de vista das súas dimensións (efectividade, eficiencia, atractivo, tolerancia a erros e facilidade de aprendizaxe).

Pautas de deseño de interfaces

Cando se traballa nun proxecto de desenvolvemento dunha aplicación multiplataforma, hai que ter en conta, en todas as **fases do seu ciclo de vida**, a creación das interfaces que permitirán a interacción coas persoas.

No deseño dunha interface hai que ter en conta algúns principios básicos:

- **coñecer ás persoas usuarias:** as súas capacidades, necesidades funcionais e evolución.
- **minimizar a memorización:** utilizando selección en lugar de entrada de datos, usando nomes en lugar de códigos, facilitando o acceso á axuda, etc.

- **optimizar as operacións:** permitindo a execución rápida de operacións ás persoas usuarias.
- **permitir erros:** ofrecendo mensaxes de erro comprensibles, deseñando para evitar erros comúns, permitindo accións reversibles e garantindo a integridade do sistema en casos graves.

Tamén se terán en conta principios básicos de deseño gráfico:

- **principio de agrupamento:** organizar o espazo visible en bloques separados de controis similares e con un título por cada bloque.
- **principio de visibilidade e utilidade:** os controis usados frecuentemente deben ser visibles e facilmente accesibles.
- **principio de consistencia intelixente:** utilizar unha distribución da información similar para funcións similares, para habituar ás persoas usuarias a encontrar a información nos mesmos lugares en situacións similares.
- **principio de economía do deseño:** omitir calquera elemento que non aporte ningunha información.
- **principio de cor como suplemento:** utilizar as cores con medida para destacar a información.
- **principio de redución do desorden:** resume os principios anteriores. Se só son visibles os controis máis usados, agrupados con sentido, con uso minimalista da cor e sen elementos superfluos, entón terase unha interface atractiva e funcional na que o desorden e a arbitrariedade estarán reducidas ao mínimo.

Deseño da estrutura das interfaces

A estrutura das interfaces refírese aos elementos que a compoñen e a súa posición dentro da mesma.

Os elementos máis habituais que compoñen unha interface son: menús, ventás, cadros de diálogo e atallos de teclado.

É importante coñecer a contorna de execución das interfaces deseñadas. Así, non será o mesmo deseñar un menú para unha aplicación de texto que para unha aplicación interactiva a través de pantallas táctiles.

Aspecto das interfaces

O aspecto da interface ten que ver coas **cores** usadas no deseño, o tipo de **fonte** empregada, as **iconas** e a **distribución** destes elementos.

Este tema moitas veces é realizado por deseñadores, aínda que ás veces son as persoas desenvolvedoras de software as que teñen que levar a cabo o deseño.

A distribución dos elementos nunha interface debe permitir un uso e unha aprendizaxe óptima da mesma.

Os obxectivos da distribución dos elementos son os seguintes: lograr que a interface sexa fácil de usar, fácil de aprender, segura, fiable e efectiva á hora de levar a cabo as accións necesarias para unha aplicación. Ademais, debe ser consistente.

Elementos interactivos

Son aqueles que permiten a navegación pola aplicación, mostran as alternativas a escoller e ofrecen a posibilidade de executar algunhas accións.

Entre os elementos interactivos encóntranse os botóns, radiobuttons, checkboxes e as listas despregables.

Outras pautas de deseño

Á hora do deseño dunha interface hai que ter en conta os elementos e regras vinculados á xestión de datos, tanto no relativo á súa presentación como no relativo á súa manipulación e destrución. É dicir, a vinculación dos datos a unha base de datos.

Outros elementos a ter en conta son a secuencia de control da aplicación, o aseguramento da información e outras pautas específicas para o deseño de interfaces con elementos multimedia.

Presentación dos datos

Á hora de escoller a forma de presentar os datos, hai que ter en conta o tipo de pantalla que será usada coa aplicación.

Alguns consellos para deseñar a presentación de datos:

- mostrar información concisa e clara.
- permitir unha navegación fácil.
- deseñar os contidos usando unha estrutura piramidal (as ideas importantes ao principio).
- usar unha linguaxe próxima ás persoas usuarias.

Secuencia de control da aplicación

A secuencia de control da aplicación refírese ás accións que as persoas usuarias levarán a cabo co software e a lóxica de execución do mesmo. É dicir, se unha persoa pide executar unha operación que implica diversas accións coa base de datos, haberá que establecer un control da secuencia de operacións que se levarán a cabo para garantir que a información se manteña consistente na base de datos.

Aseguramento da información

Para asegurar que os datos están dispoñibles cando se necesiten é necesario establecer un proceso de mantemento e utilización dos mesmos que non comprometa a integridade. Deben utilizarse sistemas para asegurar a dispoñibilidade da información, o que minimizará o risco de quebra do sistema informático. Estes poden ser sistemas de reserva ou de emerxencia que fagan copias de seguridade de maneira automática e regular.

Outro método para asegurar a integridade da información consiste en establecer sistemas para evitar o acceso directo aos datos por parte de persoas non autorizadas. Estes sistemas poden consistir nun sistema de permisos de acceso aos datos e o uso de contrasinais.

Tamén se deben minimizar os erros humanos que se poden producir ao xestionar a información por parte das persoas. Para iso, deben facerse as comprobacións necesarias do formato dos datos introducidos e pedirse confirmación ao realizar certas operacións críticas como borrado de datos.

Aplicacións multimedia

As aplicacións multimedia son aquelas que, ademais de texto, poden presentar elementos como son, imaxes, animacións, vídeos, etc.

O uso de elementos multimedia nas interfaces mellora a retención da información que se presenta.

No deseño de contidos multimedia debe manterse un equilibrio entre os contidos que aproveitan as posibilidades de hipertexto e multimedia e os que non. En caso contrario, o contido da interface será caótico debido a un exceso de elementos multimedia.

Java Swing - Look And Feel

As interfaces gráficas son algo fundamental cando se programa pensando nas persoas usuarias e na experiencia que poden ter ao manexar unha aplicación (usabilidade).

Neste apartado veremos unhas opcións que ofrece Java para cambiar o aspecto visual que mostran por defecto as aplicacións.

O Look And Feel refírese á aparencia dunha aplicación, como se presentan os diferentes compoñentes, botóns, campos de texto, combos, ...

A arquitectura de Swing está deseñada para poder cambiar o “look and feel” (L&F). “Look” refírese ao aspecto dos compoñentes e “feel” refírese á forma en que se comportan.

A arquitectura de Swing permite múltiples L&F separando cada compoñente en dúas clases distintas: unha subclase **JComponent** e a correspondente **ComponentUI** subclase. Por exemplo, cada JList terá unha implementación concreta de ListUI.

É responsabilidade de cada L&F de proporcionar unha implementación concreta de cada subclase **ComponentUI**. A creación deste compoñente é manexada de forma automática por Swing. As persoas programadoras non teñen que preocuparse de crear estas clases.

Look and Feels dispoñibles

En cada sistema hai unha serie de Look and Feels instalados e tamén se poderán instalar outros novos.

A lista dos L&F instalados dependerá do tipo sistema operativo: Windows, Linux, MacOS.

Algúns Look and Feels dispoñibles:

- **CrossPlatformLookAndFeel**: é o L&F de Java, tamén chamado “Metal” e que ten o mesmo aspecto en todas as plataformas. É parte da API de Java e é o usado por defecto se non se indica o contrario.
- **SystemLookAndFeel**: neste caso a aplicación usa o L&F nativo do sistema operativos. Este L&F determínase en tempo de execución.

Poden usarse outros L&F ademais dos proporcionados por Java. Poden importarse librerías co L&F a través dun ficheiro jar.

Exercicios

Exercicio: Crea unha aplicación que conteña un ComboBox coa lista de Look and Feels dispoñibles no teu sistema.

Para facer o exercicio necesitarás:

static [UIManager.LookAndFeelInfo\[\]](#) [getInstalledLookAndFeels\(\)](#)

Este método devolve un array de LookAndFeelInfos que representa as implementacións de LookAndFeel dispoñibles actualmente.

Ao ComboBox da aplicación debes engadir o nome (getName) de cada LookAndFeelInfo devolto polo método anterior.

Exercicio: fai que cada vez que se seleccione un LookAndFeel diferente no comboBox, cambie o Look and Feel da aplicación. Engade algúns compoñentes máis á interface para ver como se comportan ao cambiar o Look And Feel.

O Look and Feel pode establecerse usando:

static void [setLookAndFeel\(String className\)](#)

é dicir **UIManager.setLookAndFeel(lookAndFeelInfo.getClassName());**

Como a interface xa está renderizada, para actualizala unha vez establecido o Look And Feel, hai que executar as seguintes instrucións:

SwingUtilities.updateComponentTreeUI(this);
pack();

Fixarse no método main que xera automaticamente NetBeans:

```
try {
    for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
        if ("Nimbus".equals(info.getName())) {
            javax.swing.UIManager.setLookAndFeel(info.getClassName());
            break;
        }
    }
} catch (ClassNotFoundException ex) {

    java.util.logging.Logger.getLogger(PantallaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE,
    null, ex);
} catch (InstantiationException ex) {
    java.util.logging.Logger.getLogger(PantallaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE,
    null, ex);
} catch (IllegalAccessException ex) {
    java.util.logging.Logger.getLogger(PantallaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE,
    null, ex);
} catch (javax.swing.UnsupportedLookAndFeelException ex) {
    java.util.logging.Logger.getLogger(PantallaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE,
    null, ex);
}
```

Exercicio: busca e descarga un novo Look And Feel para Swing. Copia o ficheiro jar descargado dentro dunha carpeta chamada **lib** no teu proxecto e engádeo como librería do proxecto. Configura a túa aplicación para usar o novo Look And Feel descargado.

Non esquezas de executar os métodos

SwingUtilities.updateComponentTreeUI(this);
pack();

para recargar os cambios na interface.

Referencias

Para a elaboración deste material utilizáronse, entre outros, os recursos que se enumeran a continuación:

- <https://docs.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html>
- https://ioc.xtec.cat/materials/FP/Materials/2252_DAM/DAM_2252_M07/web/html/WebContent/u2/a2/continguts.html