



Group requirements

Ronny
Charlotte Morice
Zdeněk Nový
Christophe Pincemaille
Martin Vávra

November 7, 2008

Group: Desktop environment

University: Cork Institute of Technology

Department: Department of Computer Science

Module: Human Computer Interaction

Lecturer: Paul ROTHWELL

Contents

1	Introduction	1
2	Requirements	1
2.1	Environment	1
2.2	Virtual desktops	1
2.3	File manager	2
2.4	Recycle bin	3
2.5	Widgets	3
2.6	Launching applications	3
2.7	User switching	4
2.8	Miscellaneous	4

1 Introduction

The objective of this document is to summarize the proposed user requirements for the Desktop Environment Prototype. Requirements are based mainly on the previously created competitive analyses and the individual requirements specifications of our project group members.

2 Requirements

2.1 Environment

1. **Realistic metaphor** [obligatory]
The desktop environment (DE) should provide the user with a realistic metaphor. This objective could be achieved by making the desktop environment be as similar to the real world as possible.
2. **Easy access** [obligatory]
All the objects of the desktop environment the user is likely to use should be accessible in an simple way. The more the object is important to the user the faster way exists to get access to it. Because the importance of objects depends on the user of the desktop environment, another objective that needs to be met is to make the DE highly customizable.
3. **3D environment** [optional]
To increase the realism of the DE user could be provided with a 3D setting.
4. **Drag & drop** [obligatory]
The DE should make use of the drag & drop feature as much as possible because it is very intuitive and supports the idea of direct manipulation.

2.2 Virtual desktops

1. **Number of desktops** [obligatory]
There can be more than one desktop in the DE. The user is allowed to change the number of the desktops.

2. **Default number of desktops** [optional]
The default number of desktops is four.
3. **Switching among the desktops 1** [obligatory]
The user can easily switch among the desktops. This can be done using both keyboard shortcuts (KS) and mouse. The switching among desktops is designed in the way that user is clicking on an item in a list of desktops. When there are less than four desktops then a complete set of icons representing the individual desktops is presented to the user. If there are more than four desktops the user will be offered a drop-down list of the desktops.
4. **Switching among the desktops 2** [obligatory]
The other way of switching among the desktops is displaying a zoomed-out view with thumbnails of the desktops. The user can choose one of the desktops by mouse or keyboard.
5. **Invoking the thumbnails view** [optional]
To invoke the desktops thumbnails view the user moves the mouse pointer to the corner of the screen.

2.3 File manager

1. **Following the metaphor** [obligatory]
The file manager is a part of the metaphor used. The file manager and working with documents and files should not distract the user from the mental model created by the metaphor. The user should be given a view of his documents as part of the metaphor. There is no need for icons for documents because the user manipulates the documents directly. The representation of the documents should be meaningful and close to the reality.
2. **Integration of the system** [obligatory]
The file manager is not presented as a separate application, but should be integrated into the general DE.
3. **Documents moving** [obligatory]
The user can move the documents by drag & drop.
4. **Documents copying** [obligatory]
The user can copy documents by the means of drag & drop while holding down some key. Besides this, the user can copy documents by using only KSs (e.g. pressing Ctrl+C for the file being copied and pressing Ctrl+V in the destination folder as it is implemented in Windows or Linux).
5. **Documents previewing** [optional]
When a folder is open and a document is selected, it should be possible by pressing a key to preview quickly the content of the document. This will invoke a quick preview of the content of the document (like Mac OS X preview, for example) and could be done by a graphical effect.

2.4 Recycle bin

1. **Metaphor** [obligatory]

The bin should be part of the metaphor, and so be well integrated in the whole environment. It should create a mental model for the user.

2. **Primary usage** [obligatory]

The usage of a recycle bin should be as realistic as possible. This can be achieved by using the drag & drop feature. In this way the user can move most of the documents or other objects to the recycle bin.

3. **Secondary usage** [obligatory]

The other way of moving documents to the recycle bin is to use the context menu of the object.

2.5 Widgets

1. **Affordances** [obligatory]

The user is provided with a set of simple small handy applications, e.g. clock, calendar, weather forecast, market exchange rates, post-its, TV listings, battery status, quick dictionary etc.

2. **Metaphor** [obligatory]

The widgets are also well-integrated into the DE and its metaphor. The layout and position of these widgets could be derived from the usage of their equivalents in the real world.

3. **Positioning** [obligatory]

The positioning is carried out in the means of drag & drop.

4. **Adding** [obligatory]

To add a new widget user can just drag & drop it from the widget repository to the exact place where the widget is to be displayed.

5. **Removing** [obligatory]

To remove the widget the user simply has drag & drop it in the recycle bin or to use the context menus of the widget.

2.6 Launching applications

1. **Ways of launching applications** [obligatory]

There are several ways of launching applications. Each user can choose the way that best fits his/her needs or experience. The basic user is most likely to make use of a well structured menu (based on the usage categories) which gives him/her the overview of all the applications available. The advanced user needs a fast access to launch applications.

2. **Indirect launching of applications** [obligatory]

If the user opens a document whose type is associated with some application then this application should be started.

3. **Application launching feedback** [obligatory]
When the application is launched the user gets a feedback from the system that the application is being started.
4. **Icon zooming** [optional]
Icons for launching applications support the effect of zooming in while the mouse pointer hovers over them.
5. **Dynamic menu** [optional]
A dynamic menu serves for the advanced users who know the exact application name. The users is provided with a dynamic list of all applications available and an input box. While writing an application name the list of offered applications is limited to those which contain the string typed. User click an item of the dynamic menu in any time to launch the application or can use a special key.

2.7 User switching

1. **Multi user support** [obligatory]
The system supports multiple user accounts.
2. **Session handling** [obligatory]
The operating system and also the desktop environment should provide support for three ways of switching among users. First, the user who is currently logged in logs out and then the other user logs in. In this case the session of the first user will be closed. The second way is to keep the session of the user who was logged in previously. In this case the session of the logged-in user will be saved. The information for the session state will contain the list of applications that were running in the time of logging out and the opened documents. The third possibility would work only for switching among users on one computer and would be quite similar as the fast switching feature which is implemented in Windows XP system. In this case logging out does not mean closing the running applications because they stay running in the memory.
3. **Metaphor** [obligatory]
The suspending mechanism should support the metaphor. It should appear like if the user leaves his working place or office and then come back at some time.

2.8 Miscellaneous

1. **Terminal** [optional]
The advanced user can make use of a terminal application.
2. **Simple application installation** [optional]
The installation of application should be as simple as possible. This could be achieved by the same means as in MacOS system where there is a special folder for applications in which the applications are simple dropped when being installed. Installing and uninstalling applications requires that the user has appropriate permissions.