UNIVERSITY OF YORK

Submitted as part fulfilment of the degree of
MSc in Human Centred Interactive Technologies

Design and evaluation of an interface for finding and opening documents

Author:
Felipe Erias Morandeira

Supervisor:
Dr. Alistair Edwards

Number of words 14,489, counted by TeXcount, excluding appendices.

SEPTEMBER 20, 2010

Abstract

This project describes a design for a file chooser dialog that could be used for opening files from a runnning application. This design tries to allow for the use and combination of different retrieval strategies in order to improve their effectiveness; by being launched from an application the dialog could leverage information about the user's current task. The literature review analyses the different parts in which human memory is structured and studies a wide set of interfaces for document management. All this information is then used to inform an initial design of the interface, which is then evaluated using cognitive walkthrough and redesigned to fix the issues found. A prototype was implemented which had a high fidelity interface but a completely simulated functionality; this prototype was used to carry out an experiment were ten participants were asked to perform four retrieval tasks. This experimental evaluation of the interface was successful in that all the participants were able to accurately complete the tasks. The report contains further analysis about the differences between participants and the strategies that they used. We finish by including some conclusions and ideas for further research.

Statement of ethics

The project's aims and experimental design were designed with the following principles in mind: do no harm, request informed consent and keep the confidentiality of data.

Participants were not subjected to any risk different than those normally found on every day life. They were all over 18 years old and were informed. The participants were given an experiment briefing and signed a consent statement, both of which are available in appendix B. The participants were given the appropriate information before and after the experiment, and had the right to withdraw from it at any point. The participants did not receive any incentive for their participation, and this was made explicit in their statement off consent. All the information collected has been kept confidential and is only made available in aggregate form.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

As we will see, there is a plurality of approaches to document management. Underlying them is the very nature of human memory, which is complex and multi-faceted, leading to the existence of multiple strategies and cues that can be used at a given time for retrieving a specific piece of information.

During the past decades, interactive systems have used different strategies to tackle the problem of document management. The first computers relied on users to manually load data using physical media; the management of digital documents was therefore analog to that of physical ones. This began to change with the introduction of storage hardware where the files were a logical concept rather than physical, tangible objects. A step forward in this direction was the introduction of hierarchical file systems, which were originally used by the Multics operating system (1969) and have been used ever since by general purpose systems such as UNIX. In a hierarchical file system information is contained in discrete files which are placed inside a tree of directories.

The Apple Lisa (1983) introduced a visual metaphor to represent this concept: folders, such as could be found in any office at the time. Decades later, we are largely using the same basic metaphors and representations that were introduced back then. And for a good reason: as it will be explained in the literature review, the usage of folders has cognitive and organisational benefits that a good interface might strive to preserve.

Apart from navigating the file hierarchy, the other popular approach to document retrieval is searching. This works very well in the Web, but the literature shows that in the personal archive it is usually only used as a last resource when navigation has failed to locate the item. Additionally, usually the search facilities are linked to the desktop, so the searches are performed on the whole system. Searching might benefit from being tailored to the current task that the user is performing; a possible way to do this might be to link the search with the current application, integrating it into the file chooser dialog that is used to open documents from within the application

Another approach is the use of journals that detail the user's activity over time and that allow for self-reflection. Following a similar approach to the one outlined for search, the user might benefit from having a journal that is focused on the task that he is performing.

This organisation of files in folders might not be the best possible solution for some kinds of data. Some applications (e.g. music players, photo managers) implement their own user interfaces to manage documents in a way that is tailored to the task at hand. For instance, a music player might allow the user to navigate the music library by genre, artist or album. Since these solutions are specific for each application, it might be preferable to have a solution that could be used across applications so that they share the same rich metadata and use a consistent interface for browsing and retrieving documents.

1.2 Proposed solution

This project describes an interface for accessing documents that allows for different search strategies and that tries to adapt to the activity that the user is performing. These strategies can depend on the specific kind of file that the application intends to manipulate and on the bits of information that the user remembers about the target document. The literature review shows that people use a variety of criteria to organise and access their personal archives, and this design acknowledges this fact by trying to provide an equilibrium between flexibility and simplicity. The main goal is to allow the users to focus on selecting the best strategy or combination thereof to retrieve each particular document.

The proposed design also takes advantage of the fact that most applications already use a file chooser dialog for opening documents. As this dialog is provided by a shared library, in a realistic implementation this would mean a single point of change to substantially alter how the application manages its files from the user's point of view. The new document chooser dialog would substitute the existing one but instead of just offering a way to browse the filesystem hierarchy, the new dialog would use a variety of information that would enable different search strategies. A realistic implementation of this solution would also use a common metadata storage and would therefore be consistent between applications.

1.3 Project goals

This project proposes a tentative design for such a document chooser dialog; one that offers an interface that supports different finding strategies and that has potential to be adapted to the specific kinds of files that the application is intended to manipulate. This design leverages key ideas from human memory and human-computer interaction literature, while attempting to provide a simple and well-balanced visual aspect.

In a realistic implementation, the dialog would use a common metadata storage so the information would be consistent among applications. Moreover, some of the interface elements would adapt to the kind of files that the application intends to manipulate by using the system-wide ontologies of the metadata storage, so they would be consistent among applications manipulating the same kinds of documents. Ideally, the application should need to provide as little information as possible to the library creating the file chooser dialog, but the developers should also be able to optionally tailor the dialog to their needs.

1.4 Experiment

The experimental argument is that this solution would allow people to remain focused on the task at hand while being more effective at finding and using information than existing solutions. In order to test this, we developed a high-fidelity prototype of the proposed used interface that was used for experimental testing. It allowed us to evaluate the design of the interface and its suitability for supporting different strategies for retrieval of information items. A realistic implementation would be able to use the participant's own files but, as that would be too ambitious for a project of this kind, our prototype used a static list of files that had been generated beforehand. Despite the fact that the prototype did not cover all the functionality of the design, it allowed us to get a good representation of the user interface and evaluate it. The prototype was implemented in the Python language using the GTK+ widget toolkit; this implementation is discussed in section 4.1.

The prototype was used to perform an experimental evaluation. Ten participants were asked to use the prototype to locate specific files using vague descriptions while their actions were stored in a log. Each one of them performed the same four search tasks on a list of files that had been generated beforehand and did not relate to the person's personal archive.

The most important result is that participants were able to successfully complete the tasks that were given to them, although for some it took far longer than the average: a sign that there might still be potentially confusing elements in the interface.

It has to be stressed that the functionality of the prototype was simulated in the sense that the file list did not correspond to real files in the participant's computer. Further experiments might want to evaluate the effects of employing real files, integrating the dialog in real applications and using it over an extended period of time.

1.5 Conclusions and further work

The experimental evaluation of the prototype confirmed that the design described in this report is a promising approach. Further ideas for research and design are presented in chapter 5.

Chapter 2

Literature review

2.1 Human memory

If we are to explore ways to aid humans to locate their information items, a good place to begin will be the study of human memory itself. Most of this section has been taken from [1], which is a very good primer on human memory.

There are several memory models; the one used by Baddeley was originally proposed by Atkinson and Shiffrin. According to this model, input information from the environment is kept in *sensory registers* which are capable of storing information for a very short period of time (about one-tenth of a second for visual input, about three seconds for auditory input). Most of this information is discarded right away, and what is left is kept in a short-term store. This short-term store is called *working memory*, and holds information that is essential but only for a brief period of time. Baddeley describes the working memory as composed by a visual and a verbal subsystems, with an attentional control system working on them. This working memory is used to generate conscious (non-automatic) responses to external output.

Long-term memory contains information that is stored for long periods of time. Memories can be divided according to the type of information that they hold into declarative (explicit) and procedural (implicit). Declarative memory can be further subdivided into semantic, i.e. facts independent of context, and episodic, i.e. personal experiences. Information moves back and forth between the working and long-term memories. It is stored in the long-term storage through implicit and explicit learning, and it is retrieved from there so it can be manipulated in the working memory. Retrieval can be greatly enhanced by the use of cues, both verbal and non-verbal (e.g. visual, olfactive, auditory). Stored information may suffer from fading with time and from interferences caused by other memories.

Organizing and remembering are subjective processes that depend on personal perception, experience and point of view. Human beings try to impose meaning on what they observe, using what we can recall from our experience as a guide. This might lead to error at times, but more often than not this strategy works well enough because, on the whole, the world tends to be a lawful and structured environment. Experts in particular are very good at structuring

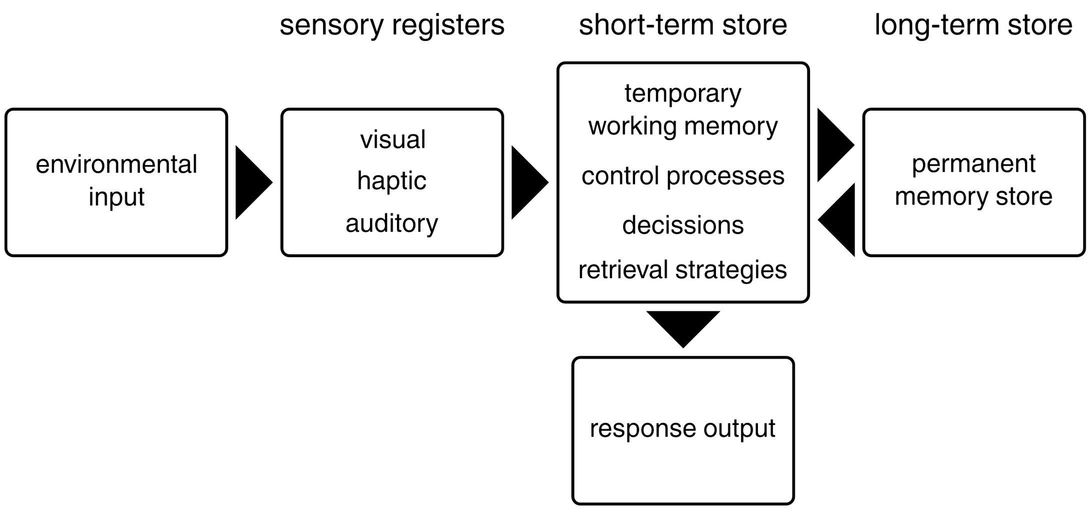


Figure 2.1: The flow of information through the memory system, as conceptualized by Atkinson and Shiffrin, and cited by Baddeley.

and making sense of complex information related to their field of expertise (e.g. chess, electronics); this might be because information that has been structured previously is easier to remember and also because experts are very good at identifying patterns and chunking[1] information. The use of techniques for organizing material (visual imagery, location. . .) can greatly improve recall, specially when several are combined; Baddeley notes several mnemonic techniques that work this way.

It is often said that one never forgets how to ride a bike: skills (procedural memories) are not easily forgotten. Baddeley describes many skills as continuous or closed-loop, in that each action provides a feedback that cues the next one; these are in contrast with discrete or open-loop skills where each action is an isolated response to a discrete stimulus.

There are two main theories for forgetting. One of them is that memories simply fade and decay with time. The other is that memories are disrupted and obscured by interference from subsequent learning: the new information supersedes the old, so the effect of interference is to make earlier memory traces less accessible and not to destroy them.

2.2 How people manage their documents

2.2.1 Finding and reminding

In 1995, D. Barreau and B. A. Nardi carried out studies of the ways users organise and find files on their computers [2]. Participants used different environments that were popular at the time: DOS, Windows, Macintosh, OS/2.

[1]A strategy for making more efficient use of short-term memory by recoding information into items (“chunks”) with a higher information content.

Despite the different operating systems and the limitations of technology at the time, this paper is interesting because it focuses on how information is used. The researchers found a number of consistent behaviours among a varied set of users and systems.

Users mainly preferred location-based filing and recovery of information. Search tools were used very sparsely and mainly only as a last resort when the user could not remember the location of their files. Search tools available at the time were rather crude in that they demanded that the user recalled the name of the file that they were looking for, they tended to be slow and return too many uninteresting results. The researchers theorize that the user's personal workspace had been arranged by them and therefore they felt more comfortable navigating through an structure that they themselves had created. Location-based filing also provided an important reminding function, as users grouped related files together and placed urgent items where they would be likely to be noticed. Barreau and Nardi argue that if file placement is to provide both *finding and reminding* functionalities, then a system in which each file is associated with a specific location is more useful than a purely logical system.

Users worked with three different types of information: ephemeral, working and archived. Ephemeral information is that which will only be needed for a short time; users preferred to keep this information easily visible (e.g. in the desktop). Working information is frequently-used information that will be needed during the next weeks or months, as it is related to the user's current work needs; this information was important enough to be properly organized. Archived information is that which was used in the past but now is only indirectly relevant to the user's current work; after a project was completed, users found it difficult to select which information to keep and where to place it. On the whole, users avoided elaborate filing schemes and archived relatively little information.

Barreau and Nardi point that a successful design must take into account the relative proportions of ephemeral, working and archived information, as each kind has its own characteristics and needs. The way that information is used (or expected to) determines how it will be organized, stored and retrieved. Participants did not tend to expend great energy on archiving old information or in creating elaborate filing structures, yet most systems seem to be focused precisely on those two functionalities.

2.2.2 Creating computer-based work environments

A year later, V. Kaptelinin carried out a similar empirical study of users' archiving habits [3], interviewing twelve Macintosh users about the strategies that they used for customizing their personal computer resources. It was found that they mostly organized their resources following local and spontaneous decisions, but interestingly these were not necessarily less usable than those created through consistent and conscious planning. Participants created separate folder structures to deal with different projects and there was a marked difference between information belonging to ongoing and to finished ones. As in [2], the process of closing a project and correctly storing its related files was reported as problematic and participants complained about having trouble identifying what should be done with each item. Keeping order and proper organisation in the workplace was a common problem and many users tended to accumulate a large amount of useless information that did not serve any practical purpose and that only added confusion.

2.2.3 Folders for project decomposition and planning

The usage of folders as a way of organising projects was studied in more detail by W. Jones et al. in 2005 [4]. They report that the use of folder hierarchies is often problematic because they can obscure and hide the information. Nowadays users have to deal with too many hierarchies, as they are used for files, emails, web bookmarks... Also, hierarchies are a limited representation because one element can only be in one place, which is poorly suited to represent certain collections of information.

Jones et al. interviewed fourteen participants about how they use folder hierarchies when working on a personal project. They found that folder hierarchies contain information about the items and their relationships; the process of organizing items in folders might help the user understand the information better. Folders are used as a way of decomposing a problem of planning a project. Because of the limitations of folders, additional information has to be squeezed into the hierarchy (such as extra characters to force a particular ordering). There is a tension between organisation for current use and for later re-use, which echoes the different kinds of information described in [2]. The users had problems because there was not support for re-using folder structures in new projects. The researchers theorize that personal information management and the management of personal projects might be two sides of the same coin.

2.2.4 The personal archive

The personal archive was explored by J. Kaye et al. in 2006 [5]. The researchers conducted a study of the different techniques employed by almost fifty academics to store their physical and digital documents. They uncovered motivations behind the construction of these archives that went further than simply storing things for later retrieval. It was discovered that building these archives also pursued the goals of building a legacy, sharing resources and reducing fear of loss. Personal archives play a big role in expressing and crafting one's identity, both with regard to others and to oneself. These findings were applicable for physical as well as for digital archives. The structure of the archive is determined by the fundamental question *"why archive?"*. Practical considerations are important to an extent, but these values of legacy, sharing, anxiety and identity construction can also play a part in defining the archive's structure and should be taken into account while designing it and when evaluating its success.

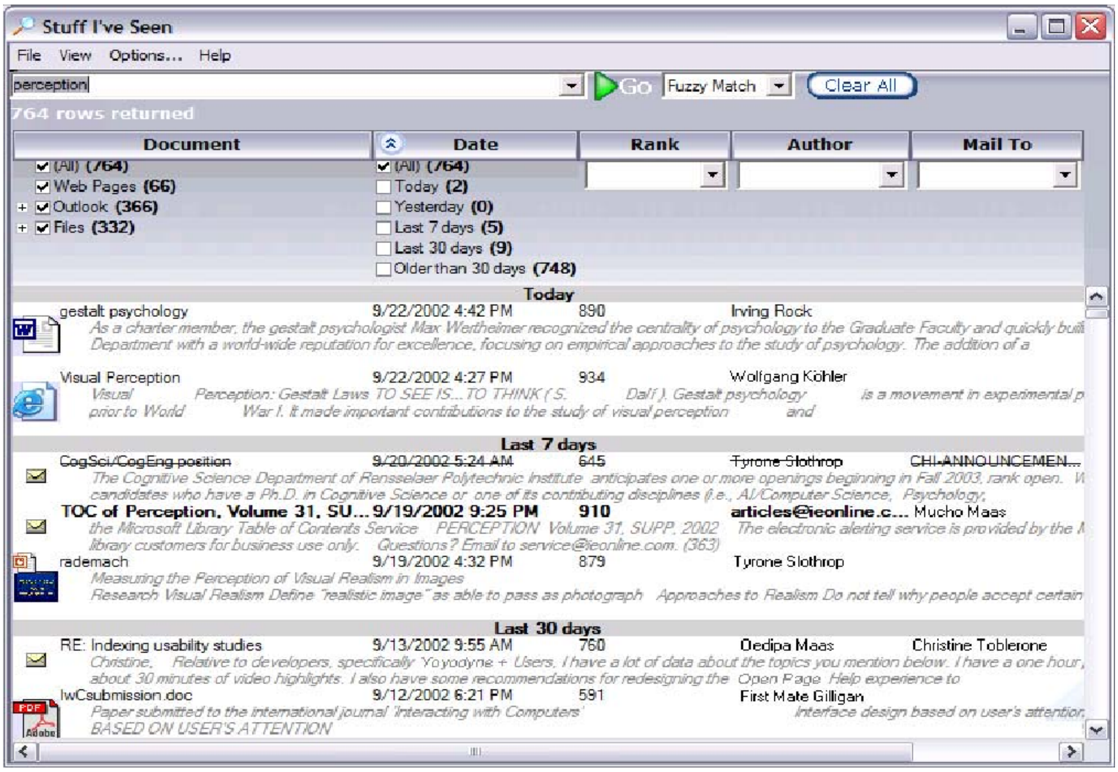


Figure 2.2: Screenshot of the Stuff I've Seen user interface, showing a set of results and options to refine the search. Source: [6].

2.3 Personal information management

2.3.1 Stuff I've Seen

Stuff I've Seen (SIS) is a system for personal information retrieval and re-use that was developed by Microsoft Research; an overview of the system is provided by S. Dumais et al. in [6]. The goal of SIS is to provide unified access to information that a person has used, regardless of its origin. SIS does that by providing an unified index with all the information that a person has seen, regardless of the format; this includes emails, web sites, documents, appointments, etc. The user interface to SIS makes extensive use of contextual cues to enrich the search interface. Dumais et al. tested SIS on over 230 participants and obtained quite positive results. People could use the system to easily find information and made extensive use of filters such as date and type; the fast interface encouraged the use of iterative refinement strategies.

2.3.2 Milestones in time

An extension to SIS was used by M. Ringel et al. to explore the use of timelines and landmarks in time for guiding search when retrieving information from personal archives [7]. The interface leverages ideas about episodic memory by annotating a basic timeline with personal and public temporal landmarks when

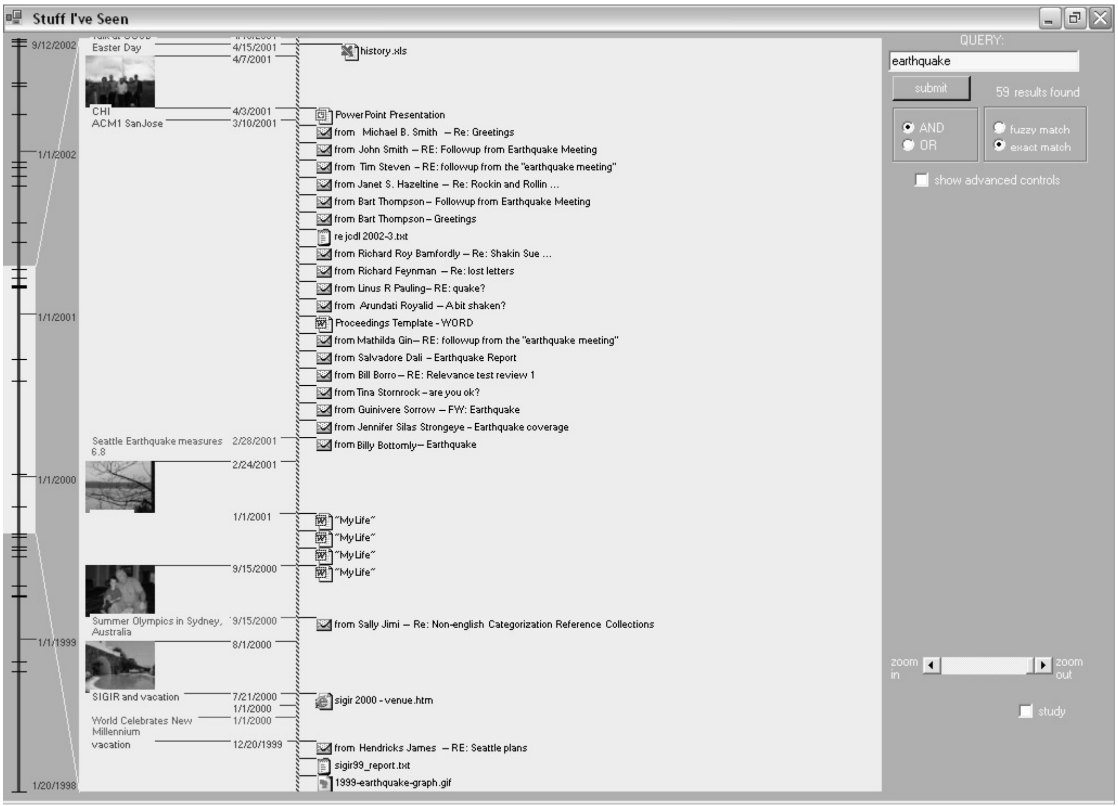


Figure 2.3: Stuff I've Seen displaying milestones next to the search results. Source: [7].

displaying the search results (see image 2.3). Its constituent elements are an overview timeline, a detailed overview with dates and temporal landmarks and a timeline backbone containing the search results. The granularity of the dates viewed depended on the level of zoom. Temporal landmarks had an associated priority to decide which ones were shown for a given level of zoom. Two kinds of landmarks were used: *public* (holiday dates, news headlines) and *personal* (calendar appointments, photographs).

Following a within-subjects design, the participants searched for emails in their system with and without the overview and landmarks. The researchers observed that median search times were significantly faster with landmarks, which showed that this way of enriching search results might point to an interesting direction for further research. The three most used attributes for searching were topic, people and time, which in this case might have been related to the fact that participants were searching for emails. It is interesting to note that personal events were considered to be more important temporal landmarks than public ones.

2.3.3 Orienteering

The iterative refinement strategies observed in [6] are consistent with the concept of *orienteering* used by J. Teevan et al. [8]. Orienteering is an information seeking strategy in which users take small steps towards their information target using partial information and contextual knowledge as a guide. This strategy is contrasted with *teleporting*, where users try to jump directly to their target, e.g. by a single search query. Orienteering and teleporting relate to the old dichotomy between search by recognition versus search by recall.

Teevan et al. studied orienteering by conducting several interviews with 15 participants at unspecified times of the day and having them report their most recent search activity. This method was similar to other diary studies used in information interaction research. Orienteering appeared to reduce the cognitive burden imposed on users by saving them from having to express exactly their information needs and by allowing them to use established habits in order to get to the proximity of the information they were looking for. Taking small steps allowed the participants to retain a sense of location, of knowing where they were, with helped in making them feel in control. They were able to get reinforcement that they were moving in the right direction, could backtrack to a previous step and felt certain that they had explored the information space completely when they were not able to find what they were looking for. The iterative process of orienteering also provided a context within which to understand the results obtained.

2.3.4 Search to eliminate personal information management

SIS was used by E. Cutrell, S. Dumais and J. Teevan to study the role of searching as a way to eliminate the need for personal information management [9]. Search engines are already a familiar means of discovering new information, and most users employ them everyday for locating relevant information in the Web. The researches explore the use of rich search tools as a possible substitute for explicit organizational structures, allowing the locating and returning to information while minimising or removing the need for structuring one's personal archive. Even in that case, the researchers acknowledge that organisational structures may support functions other than simply re-accessing information, as explained in section 2.2.

Searching for personal information is different from a search in a vast unknown collection like the Web, as people are already familiar with many different characteristics of their documents and the contexts in which they previously encountered them, and can benefit from the use of different cues. For searching to effectively replace the need to organize personal information, it needs to cut across the many possible sources of information and include all kinds of characteristics that describe the data, in order to leverage the rich associations that characterise human memory. The researchers suggest that support for many different search strategies and availability of many access routes are key benefits of a tool like SIS when compared with folder-based navigation that allows access using only a single attribute–the folder name.

The researchers carried out a study in which SIS was deployed as a voluntary download as a research prototype to Microsoft employees; there is no indication in the paper about the number of participants that took part in the experiment, although the reported results will be included here for their interest. The queries generated by the participants in the SIS study were typically short (1.59 words on average, compared to 2.16 words reported on the Web). Almost 50% of them were followed by iterations in which results were refined through sorting and filtering. This iterative process allowed users to use recognition rather than recall to find what they were looking for through a sequence of small steps.

People and time were two common ways of finding information. Over 25% of all queries included a person's name or email alias, and over 60% of the search results were sorted by date. Usage of dates highlights how what a user recalls about an item depends on context. It was found that the date users remember depends on the type of item they are looking for; as a consequence of this, the date used in the SIS interface is an abstraction–the useful date–with different date information used for different types of items. For example, for a calendar event users typically remember when an appointment happened; for Web pages, when they were visited; for photos, the date they were taken; for email, the date it was received.

2.3.5 Ranking criteria for desktop search

A detailed study on ranking techniques for desktop search was carried out by S. Cohen, C. Domshlak and N. Zwerdling [10]. They started by considering basic ranking techniques based on file features (e.g. name, size, access date. . .) and then evaluated two learning-based ranking schemes and a ranking technique based on query selectiveness. Among the basic sorting criteria, the one with the best performance was the update date, followed by the name, access date, creation date, size and content. Of the first four simple ranking criteria, three relate to a file's different dates and the fourth is the file's name; these criteria will be interesting when designing our interface in section 3. The other three sorting techniques discussed in the paper provide better performance, but they were not considered for the design because they used complex algorithms and could therefore be more opaque to users.

2.3.6 Empirical evaluation of desktop search tools

O. Bergman, R. Beyth-Marom, R. Nachmias, N- Gradovitch and S. Whittaker conducted an study on the effect that improved search engines had on personal information management [11]. They predicted that, since search is more effective and flexible for retrieval, improving the search tools should lead to a substantial increase in their usage and eventually to a preference for search over navigation; Nardi and Barreau [2] found that search was used as a last resource, and they wanted to evaluate whether this was dependent on the search technology. Also, search could solve the problems that users have with stor-

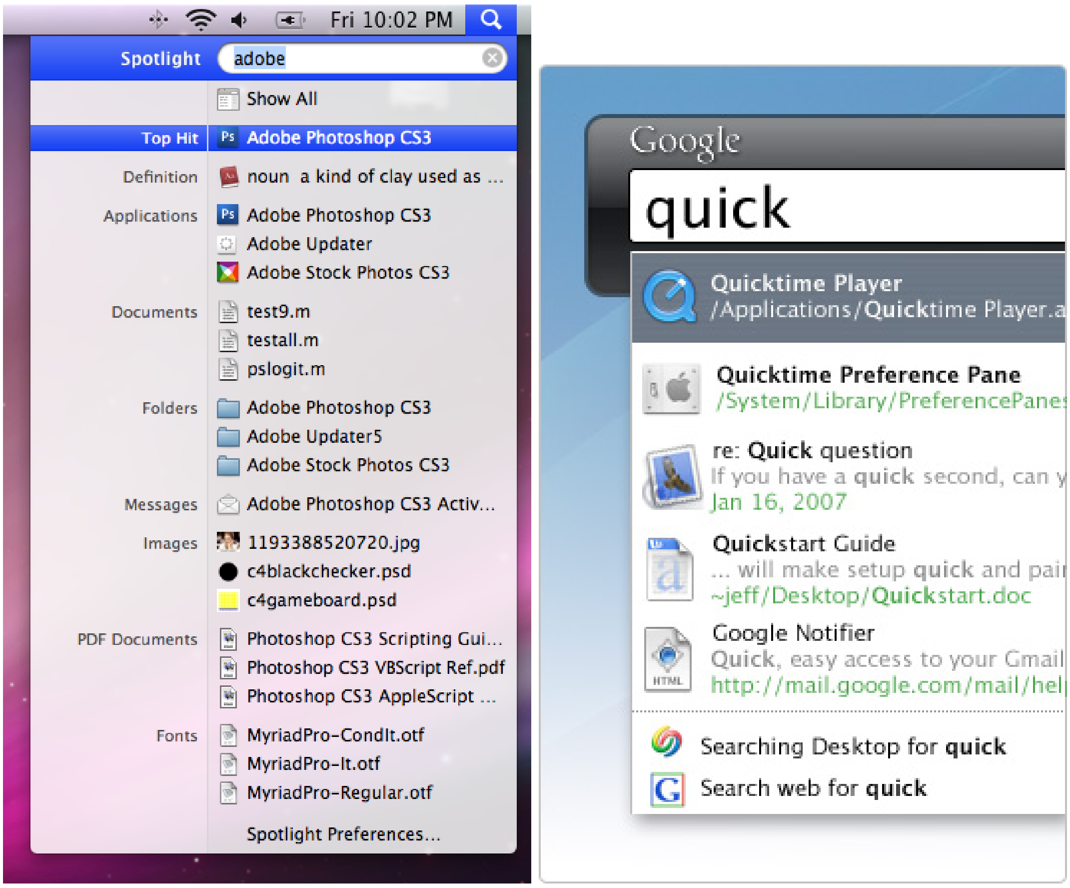


Figure 2.4: Apple Spotlight and Google Desktop Search for OSX. Source: Wikipedia.

ing files by allowing retrieval without requiring manual organisation, and thus they expected improved search tools to lead to a reduced use of complex filing strategies.

In order to study this, they evaluated the usage of desktop search tools on Microsoft Windows (Windows XP Search Companion and Google Search) and Apple Mac (Sherlock and Spotlight) using a subjective questionnaire. Their goal was to gauge the effect of improvements in search engines over the last years, such as: cross–format search, fast retrieval, user-centred design and incremental search. Google Desktop Search and Spotlight (image 2.4) have these features, the other two don't.

A valid criticism that can be raised is whether these things be accurately measured by a questionnaire rather than by direct observation. The researches acknowledge that the validity of the results depends on the participants' ability to correctly estimate their retrieval preferences and their behaviour over long periods of time. They did some pilot tests comparing self-evaluation with system logs and found little difference, but nevertheless one must be aware that the method chosen is subjective and therefore more prone to error than direct observation would be.

The results show a strong preference for navigation, with participants using it for 56–68% of file retrieval events. On the other hand, participants used search for 4–15% of events. The use of more advanced search tools did not change the general picture, and the researchers found that the effect of improving the quality of the search engine on search usage was limited and inconsistent. It does not look as if simply improving search tools would have them replace navigation. Search was used mainly as a last resort when users could not remember file location, which is consistent with the findings in [2]; it is noteworthy that there is a lapse of thirteen years between both studies. Likewise, there was no evidence that using desktop search tools leads people to change their filing habits.

The visibility and accessibility of search tools affected their estimated usage percentages. Tests with different search tools suggest that the more complex the search interface is, the less often it is used.

Bergman et al. offer a set of possible theoretical explanations for the participants' preference of navigation over search:

- *Consistency.* Using navigation is more consistent than searching, as in one case the user must remember the location of the item (which is –usually– unique), and in the other he must remember one of the several possible strategies to locate it.

- *Recognition vs. recall.* Searching requires the generation of a set of suitable search terms, demanding the recall of file names and/or other properties. Navigation is mainly based on recognition, as each intermediate step provides immediate feedback that guides the user.

- *Procedural vs. declarative memory.* The generation of a set of search terms requires the use of declarative memory; navigation also relies on procedural memory of how to get to the file and visual recognition.

- *Cognitive automation.* Browsing through a familiar self-created archive may require less cognitive attention and may be easily automated, allowing the users to remain focused on their current tasks.

- *The location metaphor.* The "location" of a file is not a real characteristic, just a convenient metaphor, but nevertheless it seems very natural because it mimics the real, physical world.

The results suggest that navigation is the preferred technique for retrieving items from a personal archive, in stark contrast with the use of search which is prevalent in the Web. In the case of the personal archive, the same user organises and retrieves the information (see section 2.2); in the Web, searching is the best approach for dealing with a massive unstructured collection of information, almost all of which the user has neither created nor consumed previously.

2.3.7 Hybrid methods

It is worth mentioning here that there exist hybrid systems for personal information management that combine search and navigation, and where this navigation is not based on folders but on alternative and richer classification methods. One such system is Haystack, developed at MIT[12, 13]. This particular system focuses on the relationship between the user and his set of personal documents and information. The user's own haystack focuses on the information with which that user interacts, gathering data about those interactions and using to further personalize the retrieval process. Haystack integrates information from multiple data sources, making extensive use of metadata, and its user interface follows a direct manipulation model.

2.4 Personal experience trace

Thorsten et al. suggest that there appears to be a need for ubiquitous self monitoring that would provide a log on what one does and experiences from one's own perspective [14] [15]. Being able to orient oneself in one's activities and with respect to one's experiences would help provide an answer to the following questions:

- "What did I do?": retrospective perspective.
- "What am I currently doing": current perspective.
- "What have I planned for the future?": prospective perspective.

Supporting individuals in answering these question would increase their feeling of being in control, make them more aware of their commitments and priorities, help them plan and carry out activities, and ease resuming and switching activities. In addition, there would be a better opportunity for self-assessment by facilitating accounting, reporting and reflecting. Fundamental to this is support for remembering and what the researchers call *mental time-travelling*. The user needs to be able to evoke elements of personal activities and experiences.

The approach followed by Thorsten et al. proposes the concept of a *personal experience trace*: a consolidation of computer-experienced events. Such a trace is constructed by logging events that denote an explicit or implicit human-computer interaction. Its purpose is to mirror what a person actually does and experiences by capturing a comprehensive, coherent and continuous picture of their activities and experiences; this picture is continuously evolving over time. It is recognized that there is a fundamental gap between what humans do/experience and what computers can sense and represent of it; it is therefore mandatory that users are and feel in control of this representation so they can correct, adapt and enrich it.

A personal experience trace consists on traces (explicitly or implicitly defined sets of events that belong together) and labels. Events are time-bound indicators of what the user does or experiences. This notion of events is based on activity theory [16]. A major task of experience trace infrastructures and

UIs is to relate events to activities and experiences. For example, by using events as start/end times to define intervals, or by using them to define activity fragments (when an activity is interrupted and later resumed). Human acting and experiencing are modeled as having intentional (conscious, controlled, goal-driven–can be labelled a priori) and unintentional (automatic, habitual, opportunistic–can be labelled a posteriori) fractions.

Thorsten et al. describe two frameworks for supporting this: *ContextDrive*[14] and *Zeitgeist*[17]. The current user interfaces for using these frameworks take the form of zooming and scrolling journals that display a user's activities and experiences along with time-expanded labels. The journals use several clues (labels, conditions, location, self-set goals...) to ease browsing. Journals UIs ease the often confusing and fragmented user experience when resuming/switching activities.

These efforts are directed at exploiting personal experience traces to inform useful personal statistics and investigate experience-trace induced notions of relatedness and relevance, providing *personal semantic technology*.

A recent interface along these lines was developed by D. Baur[18] as a tool for visualizing media and music histories. The data is taken from the Last.fm[19] service and presented from three perspectives: the user's, the individual items' (e.g. songs) and a social perspective. The goal was to provide a diary that would allow the users to recognise patterns and integrate recommendation tools. This is one of many similar visualization tools for Last.fm (more available in [20]).

Another recent interface for assisted self reflection was developed by B. Moore, M. Van Kleek, D. R. Karger and Mc Schraefel[21]. This interface took the form of an diary that generates summaries and statistical visualizations based on data collected from life tracking websites. This information is presented on a calendar that displays the activity data in textual narrative form, with the goal of making it more immediately meaningful and evocative. The researches imagine mechanisms of this kind to be integrated into common personal information management tools to help users better understand their state and improve their ability to make informed decisions.

2.5 The Web

The Web is a vast collection of unstructured data. Some of the ways used for solving the problem of retrieving information online have their origin in techniques used for personal and departmental archives. One of the first approaches was Yahoo!'s directory[22], which tried to create a very large hierarchy of categories and subcategories where Web sites were placed according to their topic and location. Search is prevalent in the Web. The Web is too vast and changing, so nowadays these location-based techniques have been superseded by search engines such as Google[23].

Another alternative for classifying large amount of content has been the use of tags, which are useful for the crowdsourcing of the structuring of big amounts of content (see [24] for a study on Flickr). Tags have also been applied

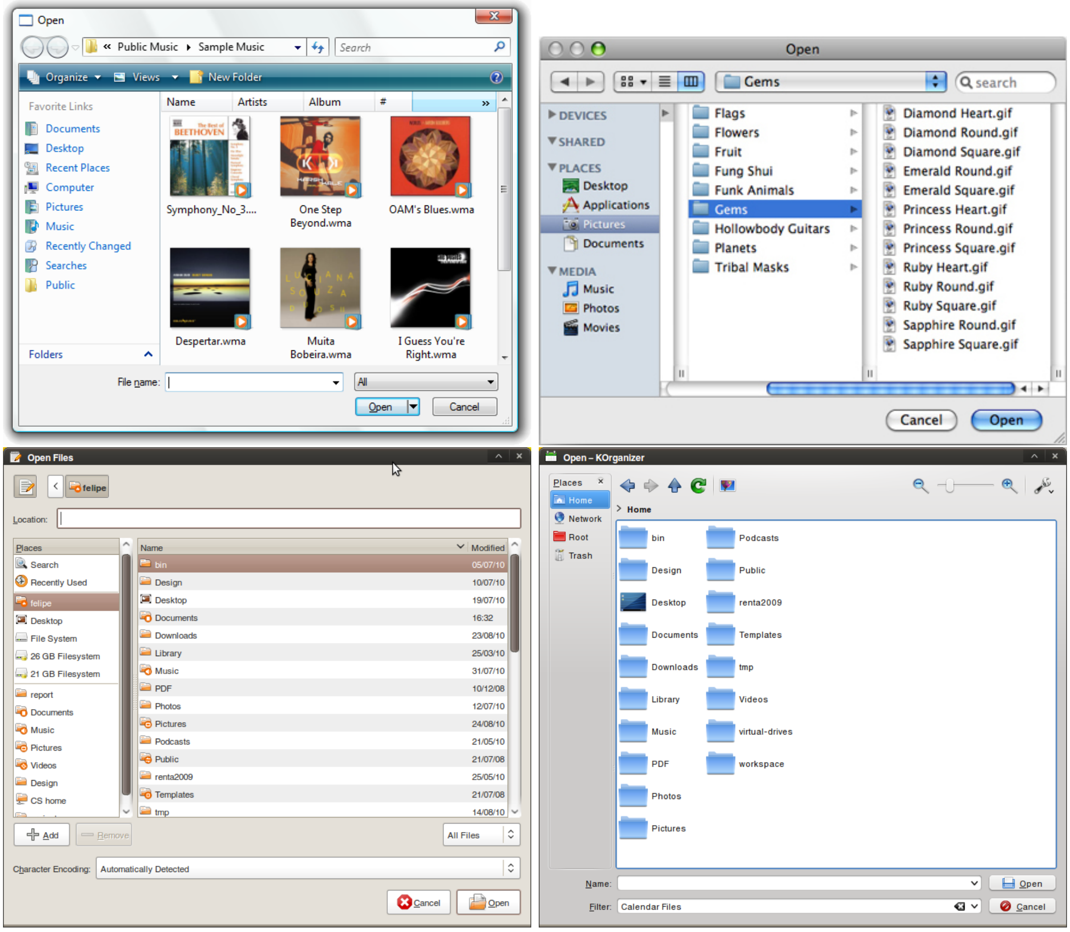


Figure 2.5: File choosers in different modern systems: Microsoft Windows Vista (source: msdn.microsoft.com), Apple OSX (source: developer.apple.com), GNOME, KDE.

successfully to personal archives for structuring certain kinds of content, such as photographs.

Facet folders, an interestins experimental user interface for the exploration of tags and other kinds of metadata, was created by M. Weiland and R. Dachselt[25]. Facet folders combine traditional folder hierarchies with metadata-based organisation, providing flexible filter hierarchies with faceted metadata. They allow the dynamic and flexible organisation of information items based on different properties: the user defines a hierarchy of such properties (e.g. time, location, activity) and the system generates a set of folders that mimic that structure. The facet folders act as a set of successive filters that organize the items into sets and subsets. The user can easily navigate and modify this structure to adapt it to the information that he is exploring and manipulating.

2.6 Existing user interfaces

A set of file chooser dialogs from four popular desktop systems is shown in figure 2.5. They are remarkable consistent with one another, as all four use navigation as their primary retrieval strategy and so the main part of the interface is devoted to displaying files and folders. They also offer the possibility of performing searches, but this functionality is kept away from the centre of the interface. All have some sort of list or tree with locations on the left side and different selectors at the top and bottom of the window.

The reason behind including these screenshots of existing implementations of file chooser dialogs is twofold. On the first hand, because it is necessary to acknowledge that there seems to be a general standard for the appearance of these interfaces. On the second, because the interface that will be proposed and discussed in the following chapters will break this consistency.

2.7 Wrap-up

The literature review provided a great amount of information on the different ways in which people organise and retrieve their information. Human memory was shown to be composed of different processing units . Long-term memory can be subdivided into different kinds of memory. Retrieval of information from long-term memory can take advantage of different cues. The memory holds different kinds of memories and is able to use different strategies for retrieval; this points towards the idea that it would be better to allow the combination of retrieval strategies than to focus on just one.

Navigation through folders relies on a series of small steps, each one of which returns a feedback that the user can employ to orient himself; recognition is extensively exploited. However, this is not the only possible paradigm and there are systems that rely on free recall through the use of search.

People store different types of information in their personal archives; some of it is ephemeral and loses its usefulness after a short period of time, other is related to ongoing projects and finally some information that was useful in the past might be archived for further use. It is interesting to note that this expectation of further use is not the only reason why information is stored; in creating the personal archive, other factors might play a part. All this information is usually placed in a hierarchy of folders; this structure often has a value in itself as a way to decompose a problem or plan a project.

The literature shows many different approaches to personal information management, from pure search facilities to experience journals. A relevant concept is *orienteering*: taking small steps towards an information goal, using partial information and contextual knowledge as a guide. This is contrasted with *teleporting*, where the user tries to reach their target in one single jump.

Some of these ideas are also applicable for information on the Web, like search and tags, but there is a significant difference between organizing and retrieving information from the user's personal archive and from a big, unstructured set as the Web.

Chapter 3

Design

3.1 Introduction and motivation

The literature reviewed in chapter 2 showed that people use a variety of strategies to store and retrieve their documents. Indeed, all the systems discussed leverage spatial, episodic and/or semantic memory to ease the filing and retrieval of information. A good design for a file chooser dialog would recognise that there exists a plurality of possible paths to a given information item. Therefore, the interface must allow for multiple strategies while remaining simple and focused. Powerful does not necessarily mean complicated.

As a sort of inspiration, Alan Cooper in "The Inmates Are Running the Asylum" [26] describes the characteristics of *polite* software: that which is personal, remembers work habits and gathers all the information that the user sends its way so it needs not ask every time for the same small piece of information. Polite software adapts to the user and not the other way around; it anticipates the user's needs and stays focused. It is responsive, well informed, perceptive and trustworthy.

Activity Theory treats interactive systems as tools that humans use to interact with their environment and with other people in order to pursue an objective. The tool is not the end: it is just a means towards a goal.

Let's begin by expressing the system's goals in the form of a *problem statement*: design a system for finding and opening files that supports the combination of multiple search strategies while remaining simple and focused.

The conceptual model for this system revolves around the use of one big list, containing all the possible files that the current application can open. This list can be explored by the user; the interface offers an overview of the list and shortcuts to different places in it. The file list supports different possible sorting criteria based on the files' characteristics. Most importantly, the results shown in the list can be sorted using different kinds of filters; modifications on these filters have an immediate effect on the list. Users search for their documents by refining the list using filters and scanning it until they find the desired file. A preview of the selected item is used so the user can confirm that it is indeed the file he was looking for.

We decided to not use personas in the design. The literature provided a detailed insight on how people manage their personal archives and retrieve their documents. Once we have this information about people in general, we did not appreciate that breaking these generalised insights into several personas would bring a clear benefit to the design process. These insights are quite conclusive and seem to hold for most people. Another argument against the use of personas in this particular case is that we did not do a previous research with concrete users and did not have a specifically defined target population of users.

3.2 Initial design

3.2.1 Functionality

Location and time play an important role in finding resources. In our daily life, whenever we have to locate an item (e.g. our cell phone) we often use our spatial memory to remember its physical location. We may also make use of episodic memory to remember the actions that we have performed with it. Additionally, we know several characteristics of the object regardless of where we have misplaced it or what we were doing with it earlier; this is another kind of memory: semantic memory. Each kind of memories might be incomplete an unable to allow us to locate the item, but together they make us able to generate increasingly approximate attempts. Each one of these attempts may also trigger the remembrance of new memories, and the process continues until the object is found (or we give up and have to ask a friend to call our cell phone number so it starts ringing under the sofa).

We begin the discussion of the desired functionality of the file chooser dialog by acknowledging that human memory can be subdivided and that the information retrieved from these different memories may be combined to form different strategies for locating a document.

The current solutions are mainly based on location or spatial memory, having the user navigate through the folder hierarchy until the desired target is reached. One of the benefits of this navigation is that the user exercises recognition rather than recall: each step down the folder hierarchy provides a feedback that might confirm that this is indeed the right path. Moreover, one can recognise these intermediate steps as part of a way or path to be followed in order to arrive at the intended destination. Almost all modern desktop operating systems use folder navigation as the standard way of organising and retrieving information.

The usage of episodic memory to identify items and tasks in time is also documented in the literature. Grouping items by the moment where they were used can be an alternative strategy that may be used to identify to establish relationships between items that are used together, maybe because they are part of the same project or relate to teh same task. Solutions such as Gnome Activity Journal, Stuff I've Seen, ReflAction Journal. . . take advantage of this to provide a temporally-oriented interface. The user of these tools can explore their past actions and identify relationships between them *a posteriori*.

Another way to leverage episodic memory is to focus not on the user's activities but on the history of the document itself. A way to do this is to allow the user to filter the results according to the origin of the files: created used this application, received from someone else, downloaded from the internet, etc. The document has a history of its own and the system and the user should be able to take advantage of his memory of his past interactions with it.

Semantic memory: search, including file type and other characteristics, such as type-dependent ones (e.g. EXIF tags)

incremental search: incrementality has advantages: user and computer do not have to wait for each other, users know when they have input enough information, users receive constant feedback of the results of the search so they can correct and modify

For this project, the goal was to develop an interface that took advantage of

Incrementality, the person uses the elements in the interface to refine their search, add filters

Place the list of items at the centre, as its the place where the effects of the user's manipulations will take place. Location on the left. Order on the right. Stability in the placement of base elements

There is an optional intermediate step between selecting a file and opening it: validating that the file is indeed the one we were looking for. An extended preview of the selected file is needed to provide confirmation that the selected item is the desired one.

Different orderings. Widget that displays a summary of the values to allow for quick navigation to a point of interest. The widget would work like a fish eye, displaying a representation of how many of the total number of items were being shown in the main list at any given time.

Concern: does this lose the physical aspect? Is this how things are in the real world? Is a filtered list a more abstract concept than folders and subfolders

3.2.2 Visual design

The visual appearance of the interface tried to focus on simplicity and equilibrium. The list of files is placed at the centre, since it is the main element of the interface and the one where the actions of the user will have an immediate effect. The different elements to manipulate the list are placed above and around it. See image 3.1 for the gradual combination of golden rectangles and squares that determined the proportions in the first visual prototype.

The most likely path followed by the eye would go from the top-left corner to the bottom-right one, as that's the case in Western languages[1]. There was a good opportunity for taking advantage of this by placing the elements in the interface according to the order in which they would be used in a typical execution:

1. filter the list,

[1]Interestingly, some systems switch the orientation of buttons for languages that are written right to left; in our case, this would mean that the "Open" and "Cancel" buttons would be placed on the bottom-left corner for e.g. an Arabic user.

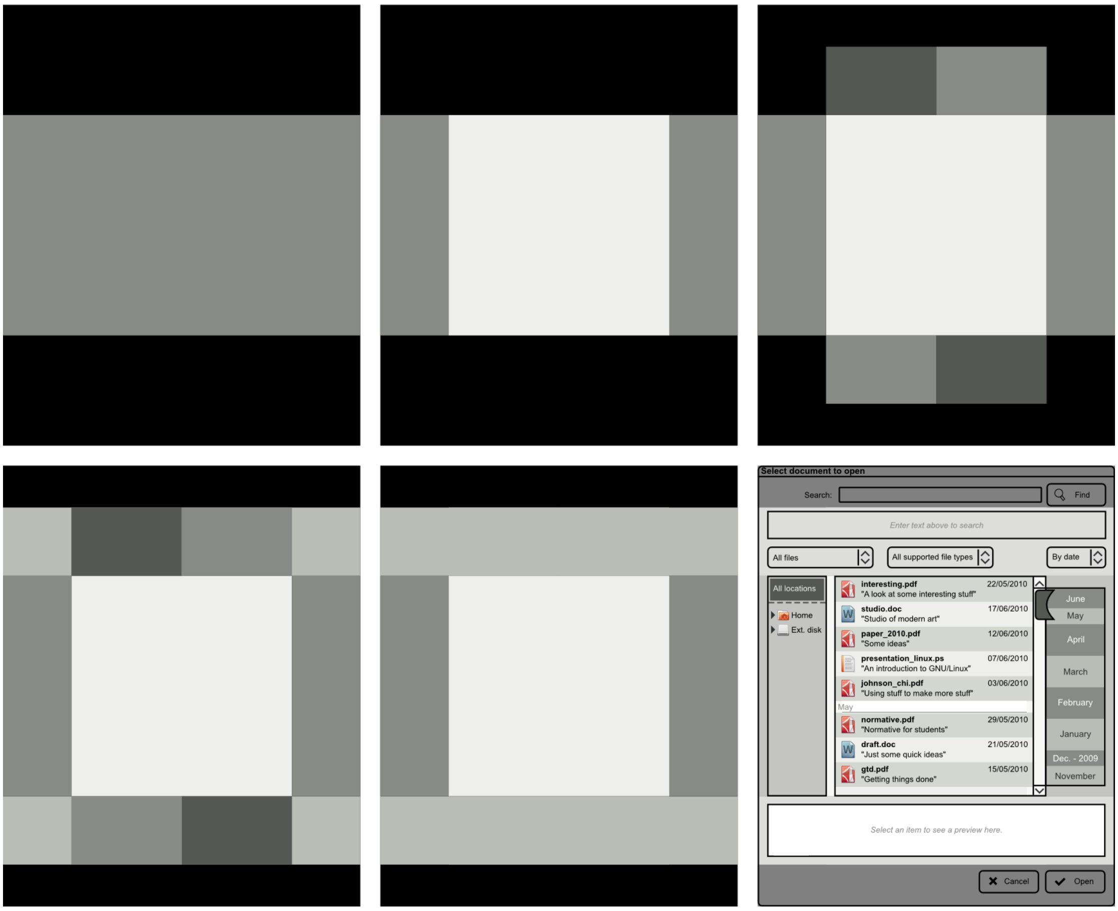


Figure 3.1: Visual design: combination of golden rectangles and squares to achieve harmonious proportions.

2. select a candidate,
3. validate that the candidate is correct (preview panel),
4. confirm the selection and close the dialog.

Thus, the filters are placed on the top-right side of the window. the file list is on the center, the preview panel is beneath it and the “Open” and “Cancel” buttons are at the bottom-right corner. This organisation tries to give a strong hint about the steps needed to open a file by establishing a visual hierarchy.

The visual design tried to be simple and focused. The basic structure was created by combining simple forms such as squares and golden rectangles, in order to achieve a harmonious visual appearance. All of the different widgets should be the ones already provided by the system, with the exception of the overview widget for which there is not a standard version and would therefore need to be implemented ad-hoc.

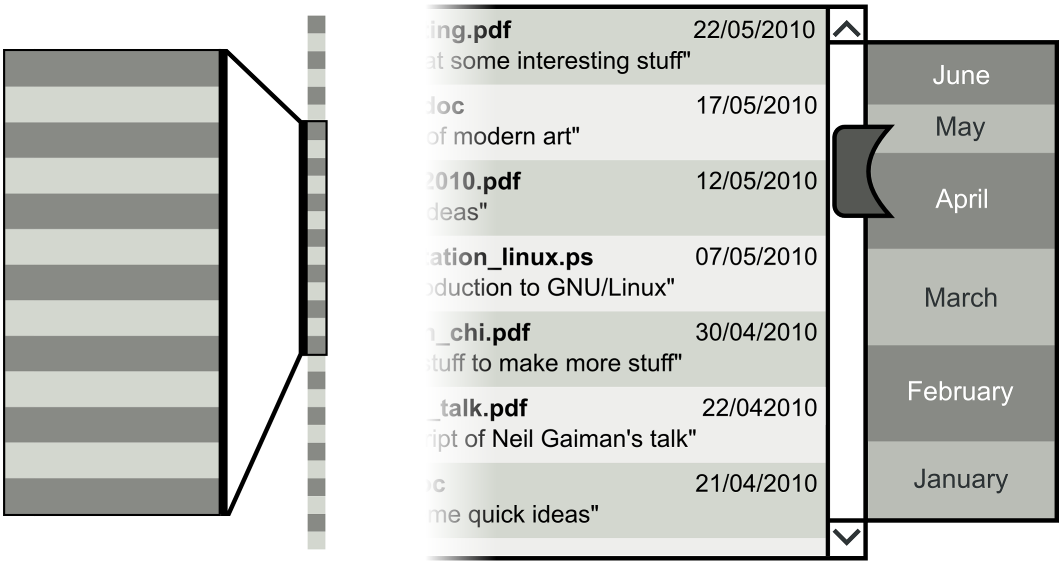


Figure 3.2: Overview widget, providing a visual summary of the whole list. To the left, the conceptual model: the widget magnifies one portion of the list so it can be manipulated. To the right, sketch showing a possible application of this concept to the file list.

3.2.3 Bringing it all together

File list

The file list is the central element of the dialog. Therefore, it is located in the centre, with the different filters surrounding it. The list is ordered by one of the following sorting criteria: file name, title, *useful* date and last use date. File name and title are the two main characteristics of a document. The useful date has the same meaning as in SIS (see [9]). The last use date orders the elements by recency. It is a reasonable assumption that a good part of the files are used more than once on close dates; therefore, many of the attempts to open a file correspond to re-use of one that has been used recently, thus justifying the existence of the last usage date sorting criteria.

These criteria provide default subsets of file and in order to help the user orient himself, it makes sense to annotate the list to indicate the boundaries between groups. In the case of file name and title sorting, the list would be subdivided by the first letter; in the case of sorting by date, the list would be subdivided according to suitable periods, such as months.

This allows for different ways of presenting the information, with an impact in how it will be accessed. For instance, ordering the results by date of use would present the user with a list of the most recent items: chances are that the desired item would often be in that group.

Visual summary

The file list can grow and became extremely long. In order to improve the sense of location and give the user a visual anchor that indicates the part of the list that is currently being displayed. The visual metaphor that has been chosen for this is a fish-eye widget that links the currently displayed subset of results with their position on the list. In order to increase the sense of location, this widget is annotated using the same subsets as the list. It is immediately apparent that a taller sector on the overview widget corresponds to more results, which provides an immediate overview of the structure of the current list. See image 3.2 for a visual explanation.

In order to provide an accelerator for advanced users, the widget can be clicked and doing that scrolls the list to the selected position in the overview. Some of the literature points that users tend to prefer to navigate through long lists of results rather than refine the search terms. This widget provides a way to accelerate these strategies and has the benefit of putting the results in context. In the particular case of sorting by date, this context is what was done immediately after and before.

Location selector

The literature showed that people tend to prefer navigation to search, so the interface contains a selector for location. The conceptual model of the interface is that the users refine and narrow the file list using filters; following this idea, the default setting is to include all locations and leave it to the user to select just one of them. Selecting a folder makes the list show only the files contained in that folder and all of its subfolders. This has potential for mistakes if there are more than one file with the same name under the current folder subtree.

Search panel

The search panel leverage semantic and episodic memory about the file that the user is looking for. Semantic memory allows the user to recall characteristics and and exploit them by selecting the file type of by using the search field. Episodic memory is used to recall details about the file's history, such as its origin:sent by a contact, created with this application, etc. Image 3.3 describes the grouping of search results by the field where the match happened, which allows the user to refine the search by selecting only one of these fields.

The searches would take place in real time, as is the case with other desktop search tools (e.g. those used by analysed by bergman et al. in [11]). Shortly before the submission of this report, Google announced *Google Instant*, a service which extends this functionality to searches on the Web (see [27]).

File preview

The last step before accepting is confirming that the correct file was selected. A file preview pane between the file list and the "Open" and "Cancel" buttons.

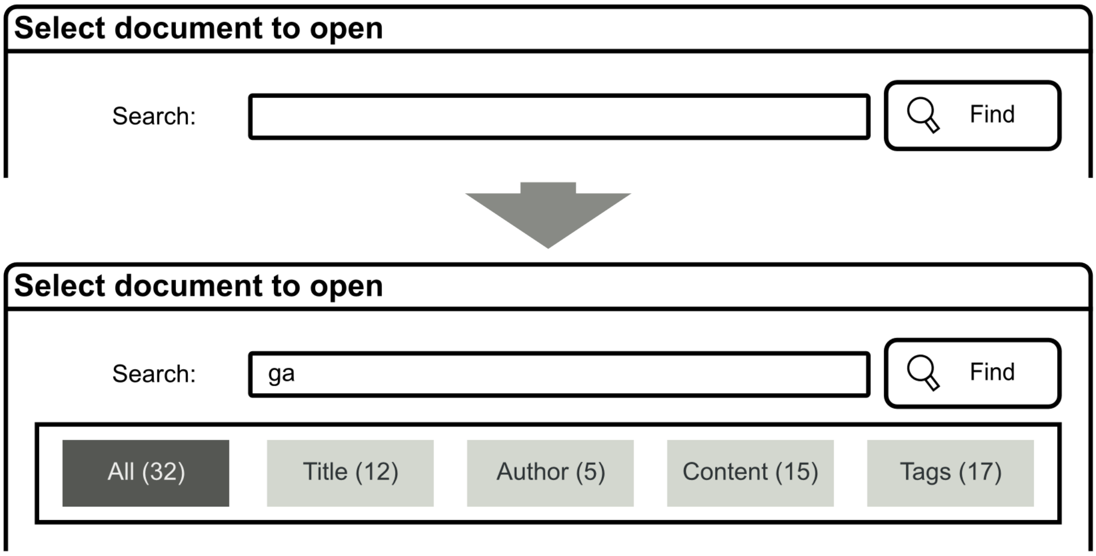


Figure 3.3: Search entry, the results are grouped by the field where the match happened, allowing the user to refine the search.

This preview shows more information about the selected file that the file list, and the information shown can depend on the kind of file; it could even include a preview in the case of multimedia files.

First visual prototype

The first visual prototype can be seen in figure 3.4.

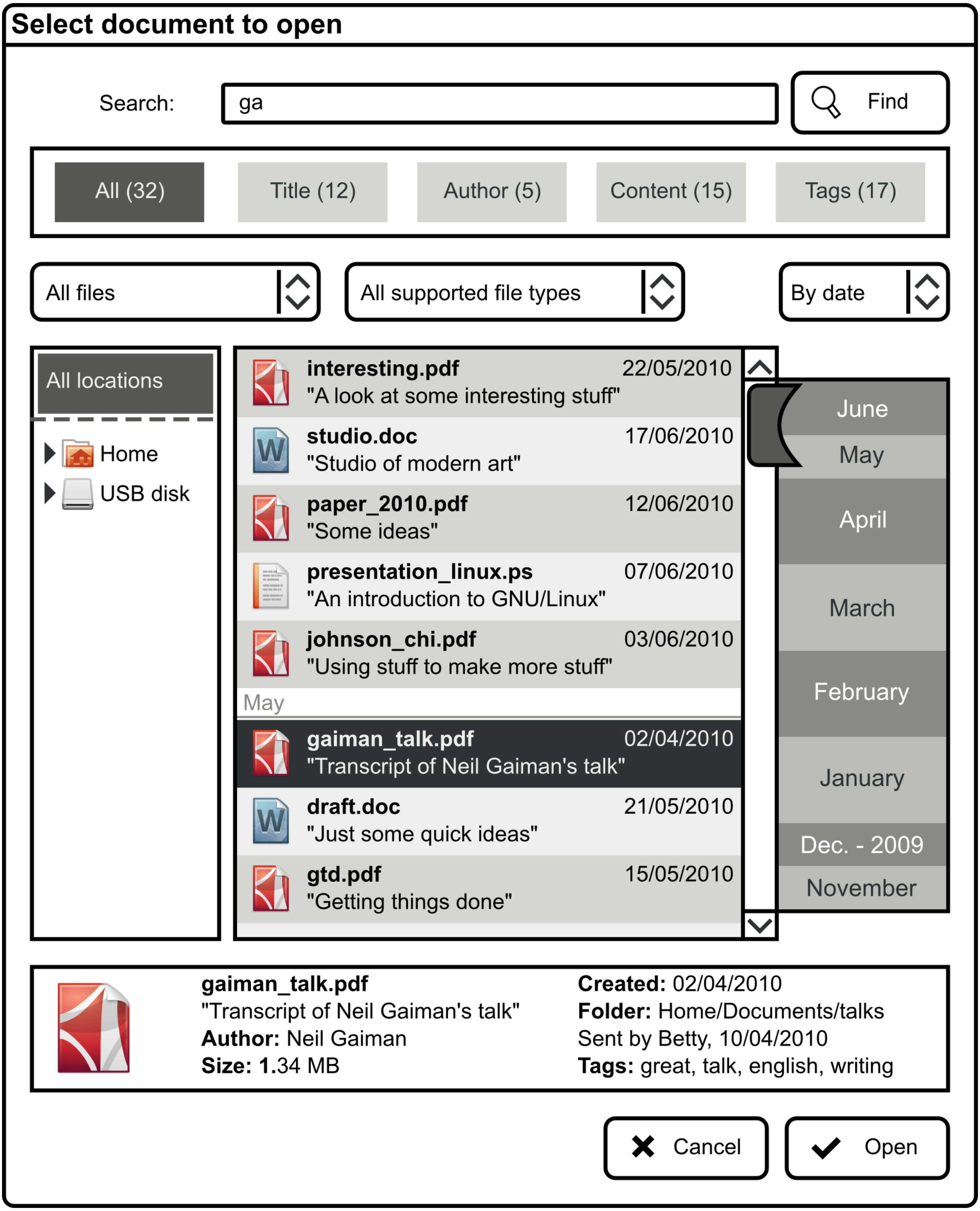


Figure 3.4: First visual prototype.

3.3 Cognitive walkthrough

3.3.1 Introduction

A *cognitive walkthrough* is a type of expert evaluation where a set of experts go through scenarios on an interface simulating the actions that a typical user would take step by step.

3.3.2 Method

Design

Each expert got a description in paper of the interaction for each of the substasks, along with a set of questions for each one. The experts got together to receive an overview of the interface and then each filled up the questionnaire on their own.

Participants

Performing a cognitive walkthrough requires the participation of HCI experts. For this one, the participants recruited were three students of the MSc. in Human-Computer Interactive Technologies at the University of York.

Material and equipment

The experts evaluated a detailed paper prototype that described the typical scenarios of the system step by step. The forms used are included in appendix A.

1. *start* : use an "Open" button to launch the dialog;
2. *sort by title* : reorder the file list using title as the sorting criteria;
3. *journal* : show the files in descending order according to their last usage date;
4. *only filetype* : display only files of a certain type;
5. *select folder* : display only files that are located in a certain folder and its subfolders;
6. *only created with this application* : display only files that were created with the same application from which the dialog was launched;
7. *sent by contact* : display only files that the user has received from a contact (e.g. through email)
8. *search* : display only files that match the keywords entered
9. *use overview* : understand the overview widget as a representation and shortcut to the current file list

10. *preview* : display more information about the selected file

11. *open file* : open the file in the application that was being used

Steps 2 to 9 were optional in that they might not be all carried out in a successful execution of the system. The experts were instructed to consider each one of those steps as carried out from the initial state of the system. Therefore, for each scenario a given run of the dialog would consist on step 1, optionally followed by zero or more of steps 2–9, followed by step 10, followed by step 11.

Procedure

The experts were given a form each, that they filled up on their own.

For each subtask, they were asked to answer the following questions. The questions were taken from [28]:

- **1 :** Will the user understand that this step/subtask is needed to reach their goal? (Yes/no)
- **1b :** Comments as to why/why not. (Open question)
- **1c :** What is the likelihood that they will have a problem understanding the need for this step/subtask? (0%, 25%, 50%, 75%, 100%)
- **2 :** Will the user notice that the correct action is available? (Yes/no)
- **2b :** Comments as to why/why not. (Open question)
- **2c :** What is the likelihood that they will have a problem noticing he availability of the correct action? (0%, 25%, 50%, 75%, 100%)
- **3 :** Will the user understand that the required step/subtask can be achieved by the action? (Yes/no)
- **3b :** Comments as to why/why not. (Open question)
- **3c :** What is the likelihood that they will have a problem understanding the correct action? (0%, 25%, 50%, 75%, 100%)
- **4 :** Does the user get appropriate feedback if they make the appropriate action? (Yes/no)
- **4b :** Comments as to why/why not. (Open question)
- **4c :** What is the likelihood that they will have a problem noticing/understanding the feedback? (0%, 25%, 50%, 75%, 100%)

	1	1c	2	2c	3	3c	4	4c
task 1	3/3	8.33 %	2/3	8.33 %	3/3	0 %	3/3	0 %
task 2	2/3	33.33 %	2/3	33.33 %	3/3	16.67 %	2/3	33.33 %
task 3	2/3	41.67 %	0/3	58.33 %	2/3	50.00 %	2/3	8.33 %
task 4	2/3	8.33 %	2/3	16.67 %	3/3	0 %	3/3	0 %
task 5	3/3	0 %	3/3	0 %	3/3	8.33 %	3/3	0 %
task 6	2/3	16.67 %	2/3	33.33 %	3/3	8.33 %	3/3	0 %
task 7	2/3	25.00 %	2/3	33.33 %	3/3	8.33 %	3/3	0 %
task 8	3/3	0 %	3/3	0 %	3/3	0 %	2/3	8.33 %
task 9	2/3	25.00 %	2/3	16.67 %	3/3	16.67 %	2/3	8.33 %
task 10	3/3	0 %	3/3	8.33 %	3/3	0 %	3/3	0 %
task 11	3/3	0 %	3/3	0 %	3/3	0 %	3/3	0 %

Table 3.1: Results of the cognitive walkthrough for each of the eleven subtasks and questions 1, 1c, 2, 2c, 3, 3c, 4 and 4c. If the question could be responded with yes or no, the corresponding cell contains the number of experts that answered yes; if the question demanded a percentage estimating the likelihood of a problem, the corresponding cell contains an average of the responses by the experts.

3.3.3 Results

Of the six experts contacted, three of them filled up the forms in appendix A. Table 3.1 summarises the results; it displays an overview of the results of the questions with a closed set of answers. This overview will be used to identify possible problems in the interface; the questions with an open answer, where the experts were invited to give their opinions, will be used to find an explanation to those problems.

The cognitive walkthrough revealed possible problems in the following subtasks:

Subtask 2: sort by title. The combobox is far from the overview widget, so the user might not discover the association between them. Sorting by title while leaving the file name in bold could be confusing.

Subtask 3: journal. The usage of the label "Journal" is confusing; the experts recommend to change it to "Sort by last usage" or something similar.

Subtask 6: only created with this application. The label seems clear, but this functionality might not be easy to discover. It should be mentioned that the combobox allows the user to select the origin of the files that will be shown in the list.

Subtask 7: sent by contact. Idem subtask 6.

Subtask 9: use overview. The experts were not sure that the usage of this element would be obvious to the user.

3.3.4 Discussion

Doing a cognitive walkthrough, even only with three experts, proved a good way to analyse the usability of the designed interface. Preparing the material forced us to reflect on the interface and how to communicate it to others. The forms used were praised by the experts as descriptive and convenient. Although each expert performed their own walkthrough, we did so together so there was a lot of informal chat going on that eased communication about the interface.

The set of problems and concerns uncovered by the cognitive walkthrough will be targeted in section 3.4.

3.4 Redesign

The cognitive walkthrough showed that the main part of the design could work well in practice but it raised doubts about some of the elements.

The combobox to select the sorting criteria for the file list was too far and separated from the overview widget, so the users might have trouble associating both elements. The solution for this was to place the combobox right above tot he overview widget. The labels in the combobox were changed to "Order by X", to make explicit the functionality of this element. Also, in order to make the field that is used for sorting more noticeable, it will be displayed in bold characters.

Another concern was about the usage of the name "Journal" to sort the file list according to their last usage date; this was therefore changed to "Order by last used".

The labels "Created with this application" and "Sent by a contact" were considered clear enough, but the default state of the combobox ("All files") was not. Since the functionality of selecting files according to their origin might not be easy to discover, the default state of the combobox was changed to "All origins".

The overview widget raised some doubts, but we took the decision to carry on with it and evaluate its usability experimentally. The labels for the months include the year as well, when the list is sorted by time.

Regarding the visual appereance of the dialog, the experts considered that the list of files was being overwhelmed by the filters surrounding it, making it look too small instead of taking the central place in the interface. There where also concerns about the filters and search elements located at the top of the window, as they made the whole interface look too complicated and might be a bit scary for some users. The experts estimated that the users would have trouble with the search panel and suggested hiding them behind an "Advanced search" label, so they would only be displayed when needed. This echoes some of the findings discussed in the literature, particularly that in general people prefer to navigate and scroll through long lists, leaving search for when other methods have failed. In order to address these concerns, the next iteration of the design moved the search elements to an *"Advanced search"* panel that would be hidden by default and only be displayed when the user clicked on a toggle button. This kept the interface simple while enabling expert users to perform elaborated queries.

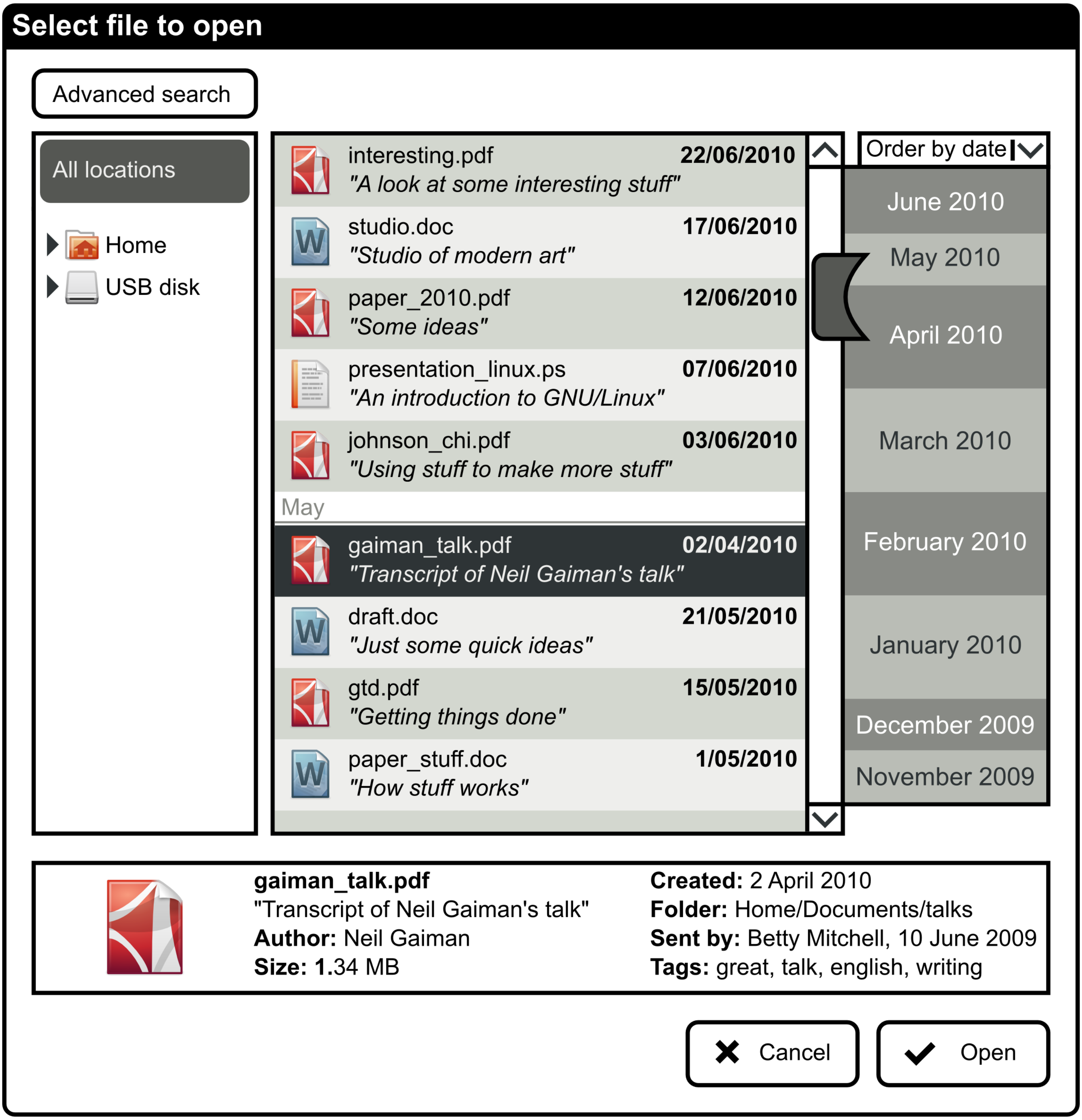


Figure 3.5: Redesign of the user interface.

Chapter 4

Experimental evaluation

4.1 Creating a prototype

Once we got a redesigned version of the interface (figure 3.5), we implemented a high-fidelity prototype to perform an experimental evaluation of this design. The prototype was built in the Python language using the GTK+ toolkit, and runs on GNU/Linux.

Two new widgets had to be developed. The first one was a custom cell renderer for displaying the items in the main file list. The second one was an overview widget that receives a summary of the current state of the file list and draws the corresponding vertical segments and labels; whenever the sorting criteria is changed, this widget simply receives a new summary and updates itself. It is linked with the file list in such a way that a click on it makes the list scroll to the appropriate position. This is not a full implementation of the fish-eye element discussed in section 3.2.3, but it is a reasonable approximation to the intended interaction.

The search overview that shows the matches per field was not implemented.

The prototype's interface is a high-fidelity one, but the actual functionality is simulated. The list of files does not represent real files in the system: it comes from a `CSV` file that has been created by a script. The file list contains three kinds of entries:

- randomly generated entries;
- solutions to the experiment's tasks;
- confounds for each of the tasks.

Our goal in creating the prototype was to evaluate the user interface. The kind of evaluations that could be performed were limited because the prototype's functionality is simulated and because it doesn't use the participant's own real files.

Figures 4.1 and 4.2 show screenshots of this prototype.

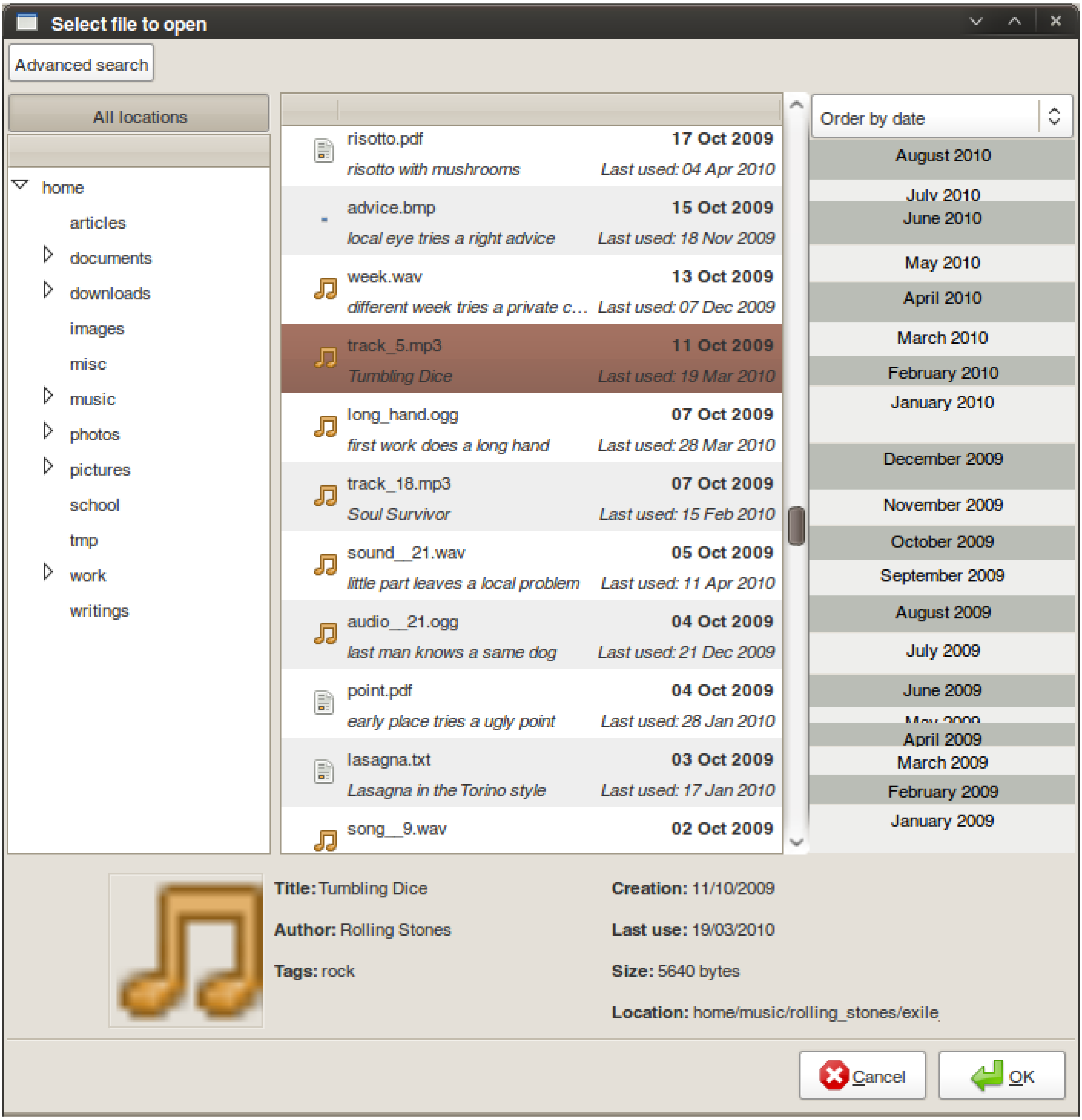


Figure 4.1: Prototype, initial screen.

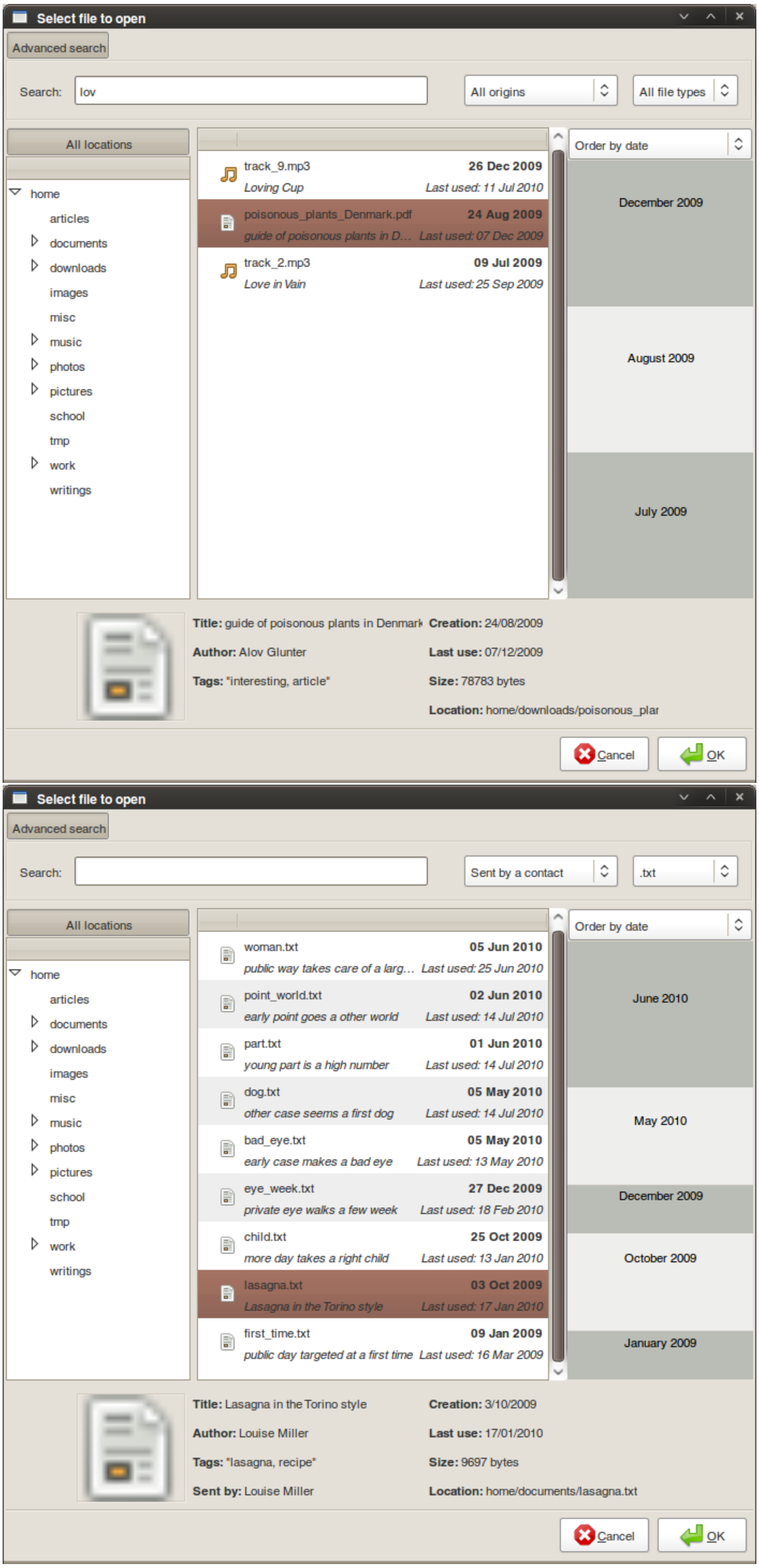


Figure 4.2: The first image shows the result of searching for the substring "lov"; the list contains two files that match in their titles and one that was sent by a person named Alov Glunter. The second image shows the result of filtering the file list to show only `txt` files sent by a contact. Note how in both cases the months in the overview widget match the creation dates in the elements in the file list.

4.2 Experiment

4.2.1 Experimental design

The aim of the experiment was to validate the design decisions outlined in the previous chapter. To this end, we performed an experimental evaluation of the prototype described in section 4.1. In order to do this, we created a set of four tasks that the participants would carry out; each task asked the participant to locate a specific file after being given a very vague description of it.

There were three variables that we were interested in measuring:

- *success rate* : whether the participant could effectively locate the target file;
- *time* : the time spent in completing each of the tasks;
- *strategies* : the way that the participant choose to locate the file.

Logs versus think-aloud

As the time in completing a task was one of the independent variables that we were interested in measuring, it was considered that the usage of a think-aloud protocol would interfere with their performance. Therefore, we made the prototype log the user interactions to a text file that could be examined after the experiment. This allowed the user to focus on the task at hand, which in turn provided more realistic results.

As an example of the kind of information logged, this is an extract from one of the log files:

```
11:38:00 2010 Launched dialog for task 1
11:38:12 2010 Selected home/writings/lit.doc
11:38:31 2010 Launched dialog for task 2
11:38:38 2010 set origin: contact
11:38:38 2010 set sort by: date
11:38:52 2010 Selected home/documents/lasagna.txt
11:39:05 2010 Launched dialog for task 3
11:39:09 2010 set search string: t
11:39:09 2010 set search string: to
11:39:09 2010 set search string: tor
11:39:09 2010 set search string: torn
11:39:12 2010 Selected
   home/music/rolling_stones/exile_on_main_st/track_7.mp3
11:39:36 2010 Launched dialog for task 4
11:39:38 2010 set folder: downloads
11:39:38 2010 set sort by: date
11:39:41 2010 Selected home/downloads/interesting/cat.pdf
```

4.2.2 Participants

Ten participants were recruited for the experiment, with the following demographic characteristics:

- 60% men, 40% women;
- 70% postgraduate students, 30% professional SW engineers;
- 5 were between 16 and 24 years old, and 5 were between 25 and 34;
- 4 of the participants were Spanish, 3 were British, 1 Filipino, 1 Chinese and 1 from the USA;
- 100% use computers almost everyday;
- half of the participants use only one desktop environment: 2 use only Apple OSX and 3 use only GNOME on GNU/Linux.
- the other five combine several systems: 4 use some version of Windows, 3 use OSX and 3 use GNOME.
- 100% are aware that their system has search facilities;
- 4 use those search facilities several times a day, 4 use them everyday and 2 use them between once a month and once a week.

4.2.3 Material and equipment

The following material was used in the experiment:

- Questionnaire (appendix B).
- Prototype (section 4.1).
- Generated list of files (355 items).

These are the four tasks that the participants were asked to complete:

Task 1: open a text document

> WORD PROCESSOR
> A few days ago I began writing a document about Yorkshire and its literature using this application. I would like to open it again so I can continue working on it. I think that I saved it somewhere in home/Documents, or maybe home/Writings?
> Your task is to locate this document. It was called lit.doc.

Task 2: open a text document

DOCUMENT VIEWER

Tonight I have a dinner with some of my friends. When I was thinking about what to cook, I remembered that another friend Louise Miller, had sent me her lasagna recipe around Fall last year. I don't remember where I saved it, but I know that I had a look at it not long ago (maybe sometime during this last spring?

Your task is to locate Louiseś recipe.

Task 3: open music file

MUSIC PLAYER

I just found myself humming the song "Torn and Frayed" by the Rolling Stones and I realised that I have not heard it in a very long time.

Your task is to locate this song so I can listen to it again.

Task 4: open a file

EMAIL CLIENT

Some days ago I downloaded a presentation in PDF with a lot of advice for taking care of your cat. I would like to send that presentation to my aunt because she is quite fond of cats. I will do this by attaching it to an e-mail that I am currently writing to her.

Your task is to locate this PDF document.

Each task was designed so it could be completed in several ways. The file list contained 300 randomly generated items, 4 answers to the tasks and 51 confound items that had been inserted in order to provide a more realistic context.

Apparatus

Seven of the experiments were conducted on a Dell XPS M1330 running Ubuntu GNU/Linux with the GNOME desktop environment.

The three participants that performed the experiment remotely used their own laptops, all of them also running GNU/Linux and GNOME.

4.2.4 Procedure

The experiments were performed individually. Each participant was briefed about the purpose of the experiment and had to agree with a statement of consent.

The preliminary questions tried to extract some information about their demographic characteristics, their usage of computers and their usage of desktop search tools.

Before beginning the experiment, the participants were given a short briefing about the interface. This was so because we were more interested in the strategies that they used than on whether they could correctly identify the interface elements on the very first time that they used it. This short brief simply identified the following elements:

- search area (containing a search field, origin selector and type selector)
- location selector
- file list
- selector of sort criteria
- overview of the file list
- preview
- confirm and cancel buttons

The experiment's script was contained in a small application that gave instructions to the participants in order to complete each of the four tasks. Each one of those consisted on giving the participants a description of a file and asking them to locate it using the *"Open file"* dialog. This involved clicking on the button below each description to open the dialog, locating the file, selecting it and clicking the *"Open"* button in the dialog. The six screens of this application can be seen in figure 4.3.

The prototype logged each action in order to create a record of the strategies followed by the participants. This technique was used instead of a think–aloud protocol because it was considered that it would be more realistic and it would not interfere as much with the participant's actions. The resulting outcome of this part of the experiment was one log file per participant, to be analysed later on.

The participants had to answer a series of questions after the experiment. The first six were inspired by the six dimensions measured by the NASA TLX[29], and used a scale that ranged from 1 ("Not at all") to 7 ("Very much so"):

- *Mental Demand:*
  Did you feel that the interface required a high level of mental effort?
- *Physical Demand:*
  Did you feel that the interface required a high level of physical effort?
- *Temporal Demand:*
  Did you feel that the interface required a lot of time to use?
- *Performance:*
  Did you feel that the interface let you successfully locate the files that you were looking for?
- *Effort:*
  In general, did you feel that using the interface required a big effort?
- *Frustration:*
  Did you feel frustrated by the interface at any point?

Procedure for remote evaluation

Participants 7, 8 and 9 performed the evaluation remotely. This was possible because the application that was developed allowed them to complete the experiment entirely on their own. They were asked to fill up the first two pages of the questionnaire, run the experiment, fill up the last page and submit both the questionnaire and the application's log file to the researcher. It has to be noted that these three participants are professional SW engineers.

The number of participants is too small to perform a complete statistical comparison between the two procedures; additionally, in this case their professional experience might introduce a bias. With more participants and homogeneous groups, we could compare the times that each group needs to successfully complete each task.

In spite of that, a look at 4.4 does not show them performing significatively different than the other participants. They are consistently among the ones with the best performance, although this might be related to their professional experience.

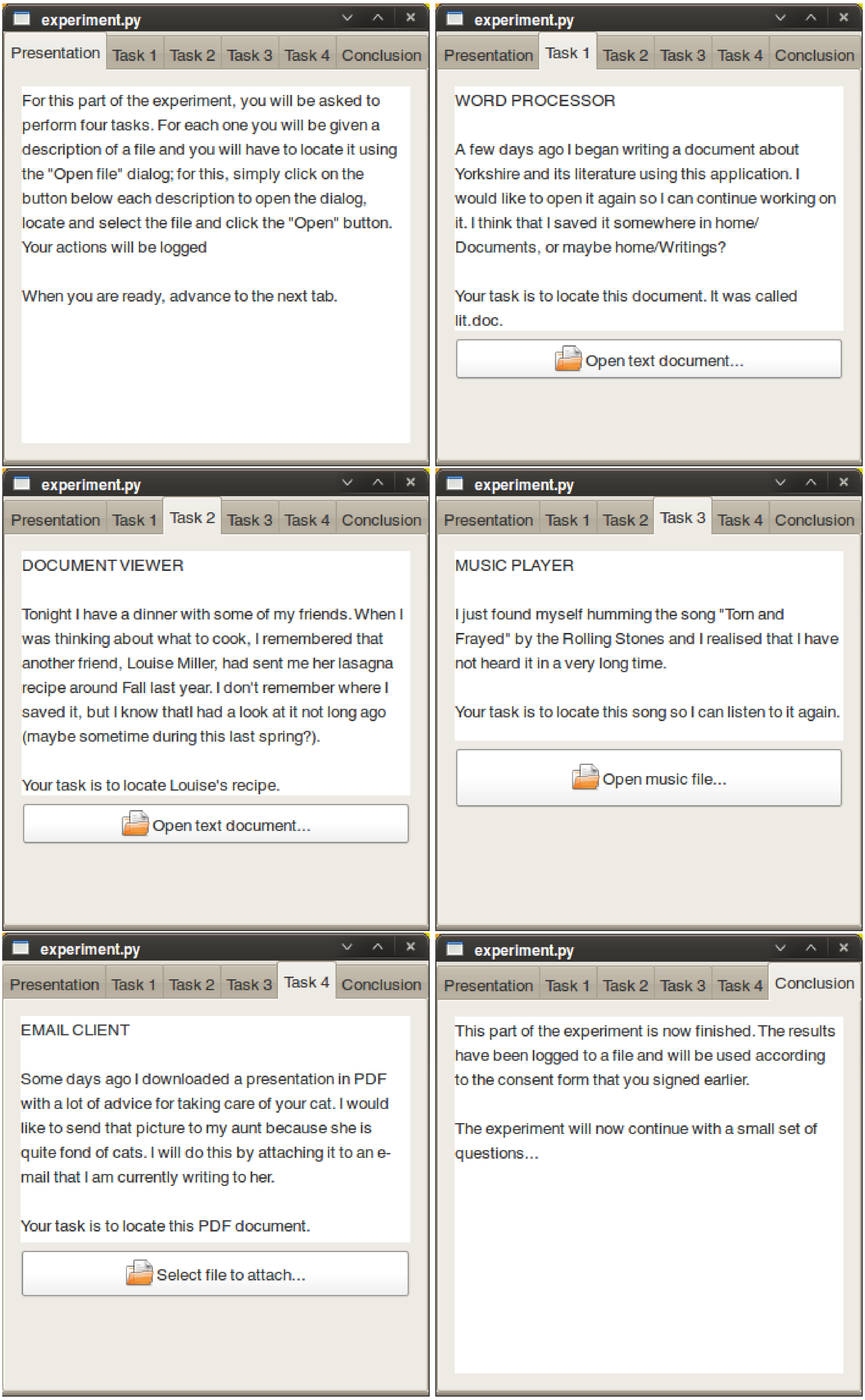


Figure 4.3: Application that guided the participants through the experiment.

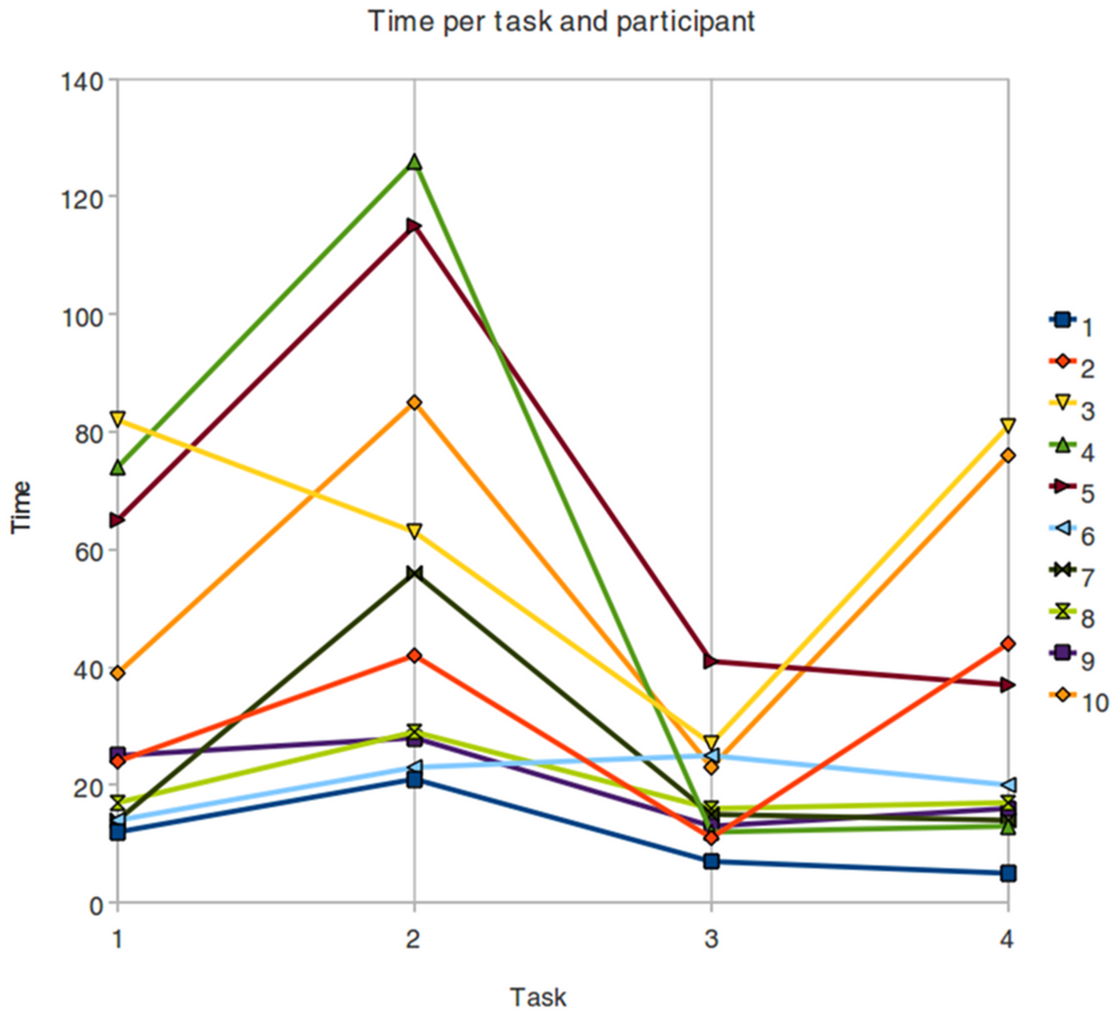


Figure 4.4: Time spent by each participant in completing each task.

4.2.5 Results

Probably the most important result is the following: the participants had 100% accuracy in identifying and locating the correct files for each task. All of them were able to use the interface to retrieve the requested items.

Figure 4.4 shows the time spent by each participant in each task. Lines have been added to help identify the performance of each participant in each task. Participants usually do comparatively better in some tasks than in others, but it seems that some of them are consistently above or under the average. Participants 1, 6, 8 and 9 performed each task in under 30 seconds.

An interesting learning effect happened with participant 4, who was among the slowest for the two first tasks but managed to perform each of the last two in under 20 seconds. The opposite case, were someone performed very well at the beginning but badly at the end, did not happen.

Table 4.1 shows the average and standard deviation for each one of the tasks. The values for each task are significatively different, which indicates that we

Task	Average	St. Deviation
Task 1	36.60	27.03
Task 2	58.80	38.32
Task 3	19.00	10.10
Task 4	32.30	26.97

Table 4.1: Average and standard deviation of the time devoted to complete each task.

should be careful when analysing the results in aggregate, as they seem to belong to different underlying distributions.

Figure 4.5 shows together the four histograms for the completion times for each task. This figure gives an intuitive idea that good strategies are available for each task; many participants found them and could find the target file in 30 seconds or less. These histograms are shown by separate in figure 4.6.

Figure 4.7 pairs the strategies followed by each participant in each task with the time it took them to complete the task. It has to be noted that two participants found a mode problem when searching after having set a location: the search applied to files in that location and not to all files, but the participants did not notice this. These two cases are labelled as "multiple" in the figure because the participants had to had to try a different strategy after encountering the problem.

The analysis of figure 4.7 shows that the participants' performance was faster when they could figure out a good strategy for finding the target file. When this was not the case, they tried out different strategies (e.g. different locations) and reverted to scanning the list, thus taking longer to complete the task.

The best strategies were different for each task. Interestingly, we can see that the fastest participants in each task got similarly short times despite using different strategies. In task 1, the fastest participants either simply scanned the default list, set the location or chose to see only files created with the current application. In task 2, the fastest used search or chose to see only files sent by a contact. In task 3, the fastest searched or set the location and then sorted the results by title. Finally, in task 4 the fastest participants combined search, file type and/or location to find the requested item.

What does this plurality of best strategies tell us? There might not be a single "best" strategy and that different people may find different solutions to the same problem. Additionally, the interface seems to be good at allowing the use of different strategies without a major penalisation. This is also a confirmation that the tasks succeeded in their goal of not being so specific that they would lead the participants into using only one strategy.

It is noteworthy that only 1/5 of the task runs made use of location. A likely explanation for this is that the wording of the tasks did not mention specific locations and the participants could not used their memory of because they were not browsing their own files.

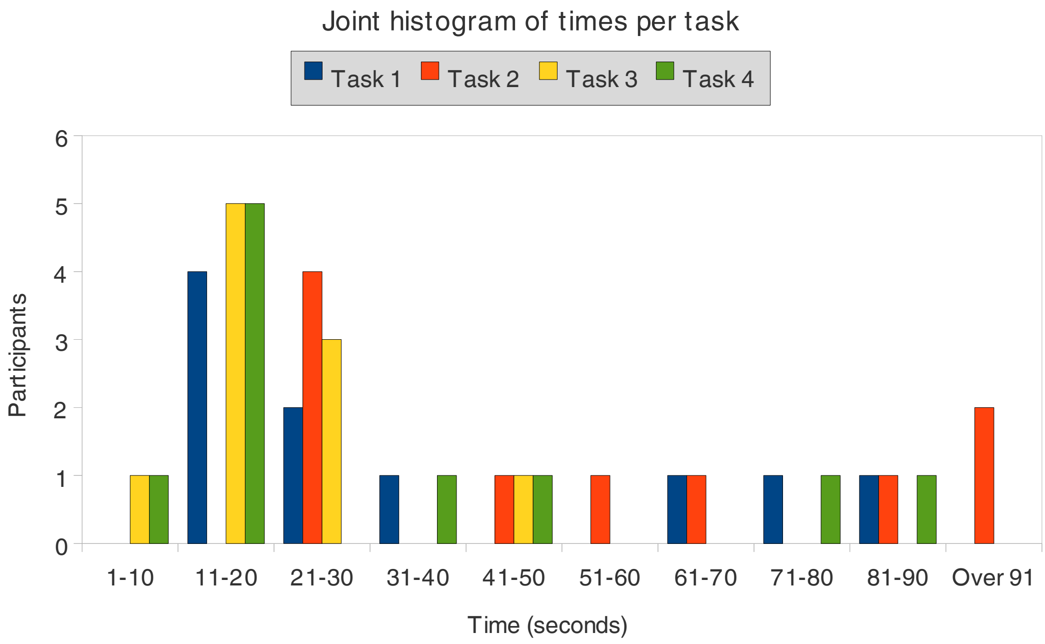


Figure 4.5: Joint histogram of completion times for each task.

The post-experiment questionnaire shows the following results:

- physical demand, effort and frustration were very low
- mental and temporal demand were low;
- performance was very high
- the participants tended to find the interface easier to use that the solutions that they usually employ.

Question	Average	St.Dev.	Mode	Median
Q1: Did you feel that the interface required a high level of mental effort?	2.50	1.43	2	2
Q2: Did you feel that the interface required a high level of physical effort?	1.30	0.67	1	1
Q3: Did you feel that the interface required a lot of time to use?	2.40	1.51	1	2
Q4: Did you feel that the interface let you successfully locate the files that you were looking for?	6.40	0.97	7	7
Q5: In general, did you feel that using the interface required a big effort?	1.90	1.45	1	1
Q6: Did you feel frustrated by the interface at any point?	1.90	1.60	1	1
Q7: How easy to use did you find this interface compared to the solution that you usually use?	3.00	1.33	2	3

Table 4.2: Statistical analysis of the time spent by the participants in completing each task. The possible answers ranged from 1 to 7, with 1 meaning "Not at all" ("Very easy" for Q7) and 7 meaning "Very much so" ("Very hard" for Q7).

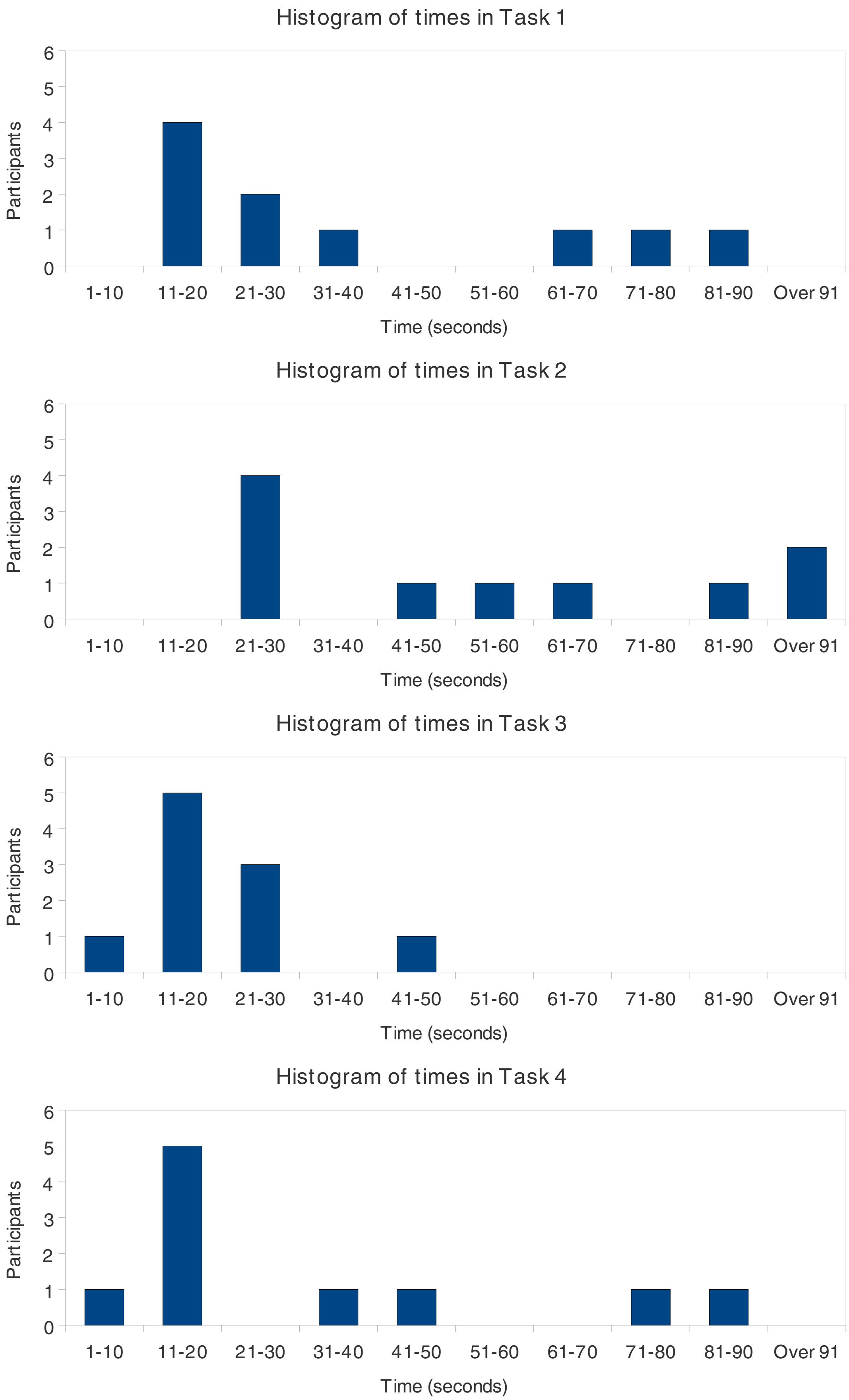


Figure 4.6: Histograms of completion times for each task.

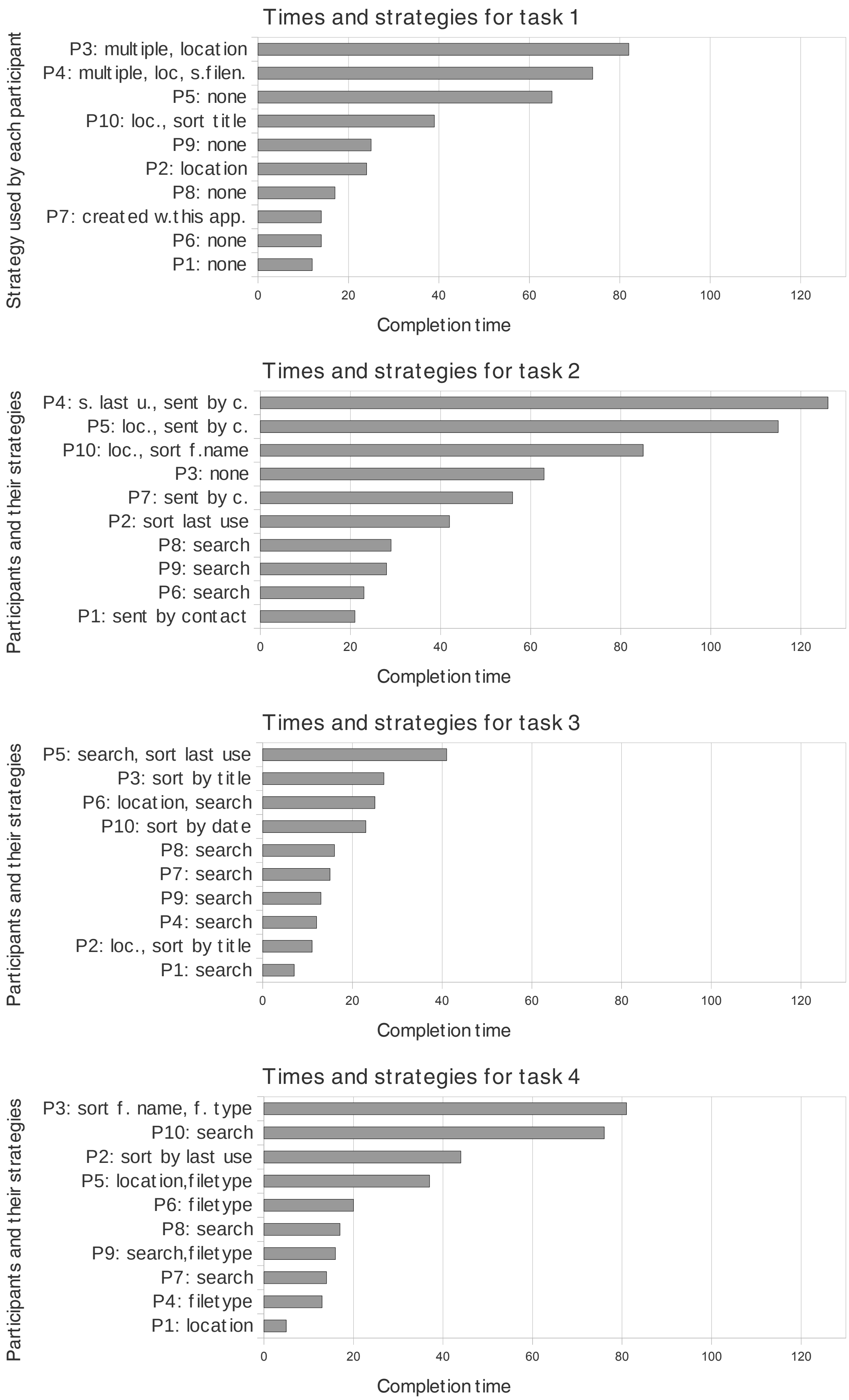


Figure 4.7: Strategies followed by each participant in each case.

4.3 Discussion

The prototype was good at testing the user interface, although the usage of a generated list of files and not the participant's personal archive limited the scope of the things that we could evaluate.

All of the participants were able to complete the tasks successfully, which is a big success for the design of the interface.

The time spent in completing the tasks was generally short, but there were many cases where the participants took a long time to complete the task. This might be explained by the limitations of the prototype, which didn't use the participant's own real files; another factor could be simply that some of the participants would need some learning and practice before being able to use the interface efficiently. On the other hand, this might point at problems in the design of the interface that made it too confusing and hard to use for the users.

There were not single best strategies for each task, as each one could be completed following different paths. The participants who could plan a good strategy were the fastest in retrieving the files; the ones who relied on browsing the file list or had to try different strategies were slower.

The evaluation detected a *mode* problem when combining filters for location and searching. This might be related to inexperience on the part of the participants, but it would be good to address this issue in the next iteration of the interface.

Chapter 5

Further work

5.1 Future research

The experimental evaluation described in the previous chapter has a relatively narrow scope. Specifically, the dialog was not integrated in a existing application, the files used were not the participant's personal files (they were not even real files) and the experiment was focused on the performance of predetermined tasks rather than on the evaluation of long term usage.

Followup experiments should fix these problems. The dialog should be integrated in real applications so it can become part of the users' real work flows. It should work of the user's personal archive and use the real data about the files. Such a high-fidelity prototype would be suitable for long term usage in realistic settings.

Further research could implement a more realistic prototype and use it on far more participants. Then we would be able to use more advanced statistical tests to find factors that might affect the user's performance when using the interface.

5.2 More functionality

This design project tries to ease the problem of file managing by providing an user interface that supported different search strategies and that could be extended to better support specific kinds of files. Further work could study ways to adapt the interface to the kind of file that a specific application intends to use.

The main idea is that the dialog should look different when launched from different applications. Contextual information would be used to support the more appropriate search strategies for the type of items that the current application intends to manipulate.

The interface offers several extension points where these extensions could be placed:

File list
the representation of items in the list should be adapted to the kind of

file being displayed in each case. For instance, in the case of images a thumbnail and some relevant EXIF tags could be shown.

Location selector
: for some kinds of files where the location of the file might not be that important (e.g. music, photographs), other kind of information might be used instead; for instance, actual physical location or tags might be used for photographs.

Sorting criteria
: four sorting criteria have been discussed (file name, title, creation date and last use); other possible criteria could depend on the particular kind of file, so for instance music files could be sorted by artist or album.

Search
: the search panel could be extended to include more options that would allow users to further refine their searches; for instance, they could select whether the search would include the files' contents or not. The list of possible origins of the file could be expanded to include e.g. "downloaded from the Internet". Participants reported that the search is not visible enough, so we could evaluate the possibility of having it displayed all the time.

Preview panel
: the preview panel should adapt to the kind of file being preview; modern systems already do that, some even going so far as to e.g. allowing the user to listen to a selected audio file from the file chooser dialog.

Chapter 6

Conclusions

We did not get the information from real would-be users of our system; instead of this, we used the wide corpus of existing scientific literature. Document retrieval is a well researched topic, so we were able to use that information to inform the design. The study of human memory describes different kinds of memories and different kinds of getting back to them through recognition or recall. These plurality of strategies for retrieval points towards the need to be able to combine them in order to increase their effectiveness.

The personal archive contains three different types of documents: ephemeral, current and archived. Most modern systems structure these documents in a hierarchy of folders; research shows that these structures may contain relevant information about, for instance, the decomposition of a problem of or the planning of a project. One can not simply do away with folders because there is an important value in structuring information.

Research in personal information management contrasts orienteering (take small steps towards an information goal, using partial feedback to orient oneself) and teleporting (reach the goal in one jump). The literature discusses many different interfaces for managing one's personal information. Journals are still largely experimental, but they show promising capabilities for self-reflection of past activities. Some of these concepts are related to Web search and the use of tags to create folksonomies, although there are important differences between a personal archive and a big unstructured collection like the Web.

We designed a file chooser that would allow for the use and combination of different retrieval strategies while remaining as simple and focused as possible. This initial design was evaluated through a cognitive walkthrough. Preparing the materials for this was a very good way top go over the different aspects of the interface. The walkthrough was a good tool for eliciting opinions about the design and identifying possible problems. Apart from the formal evaluation, the informal interaction among the experts opinions was very interesting and productive. A redesign of the interface was carried out in order to address the problems identified by the cognitive walkthrough.

This new design was used as the base for implementing a prototype using Python and GTK+. These tools proved very appropriate for fast prototyping. The implemented dialog had a high-fidelity UI but the underlying functionality was absolutely simulated. This narrowed the scope of the things that we could test, but nevertheless it served as a good tool for evaluating the main ideas behind the design.

We used this prototype to carry out an experiment where ten participants were asked to carry out four retrieval tasks using the file chooser dialog. It is interesting to note that the experiment's script was contained in a small application that presented the tasks and instructions one by one to the participant. The application logged all of the participant's interactions, so a speak-aloud technique was not necessary. Additionally, this automatisation of the experiment made it possible to perform remote evaluations.

The experiment was successful in that the accuracy was 100%: all partcipants were able to accurately use the interface to complete the tasks. For each task there were several different good techniques; participants who used one of those were faster than those who had to try different strategies or simply resorted to scanning the file list. We observed learning effects, with some participants becoming comparatively faster in the latter tasks.

This experiment had a very defined scope and it did not cover the usage of real files from the participant's personal archive, nor did it use real applications or was suitable for studying long-term usage of this system. However, the experiment was reasonably successful and it sets the direction for further research that might evaluate a more complete implementation of this design. Some ideas for extending the design are presented in section 5.

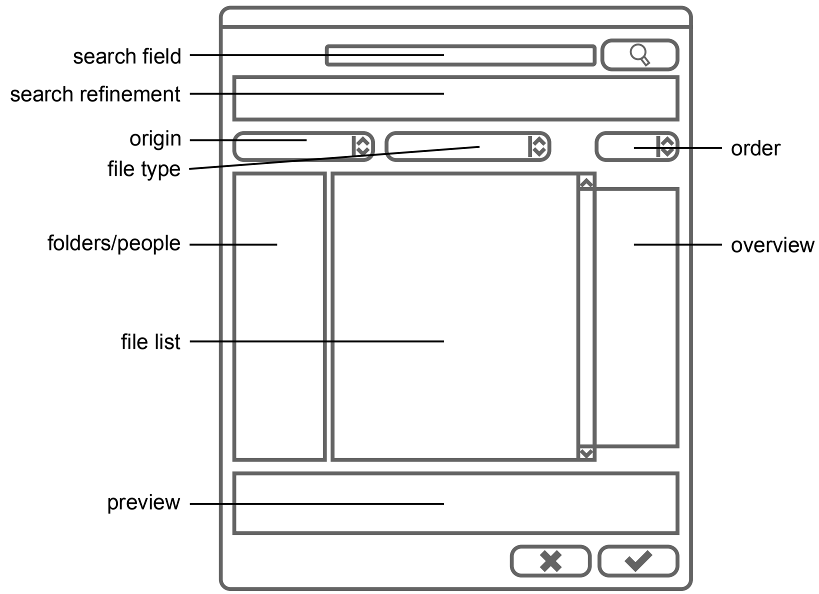
As a general conclusion, we believe that the design process followed in this project was able to provide an interesting outcome and that it could be applicable to other design problem in the future.

Appendix A

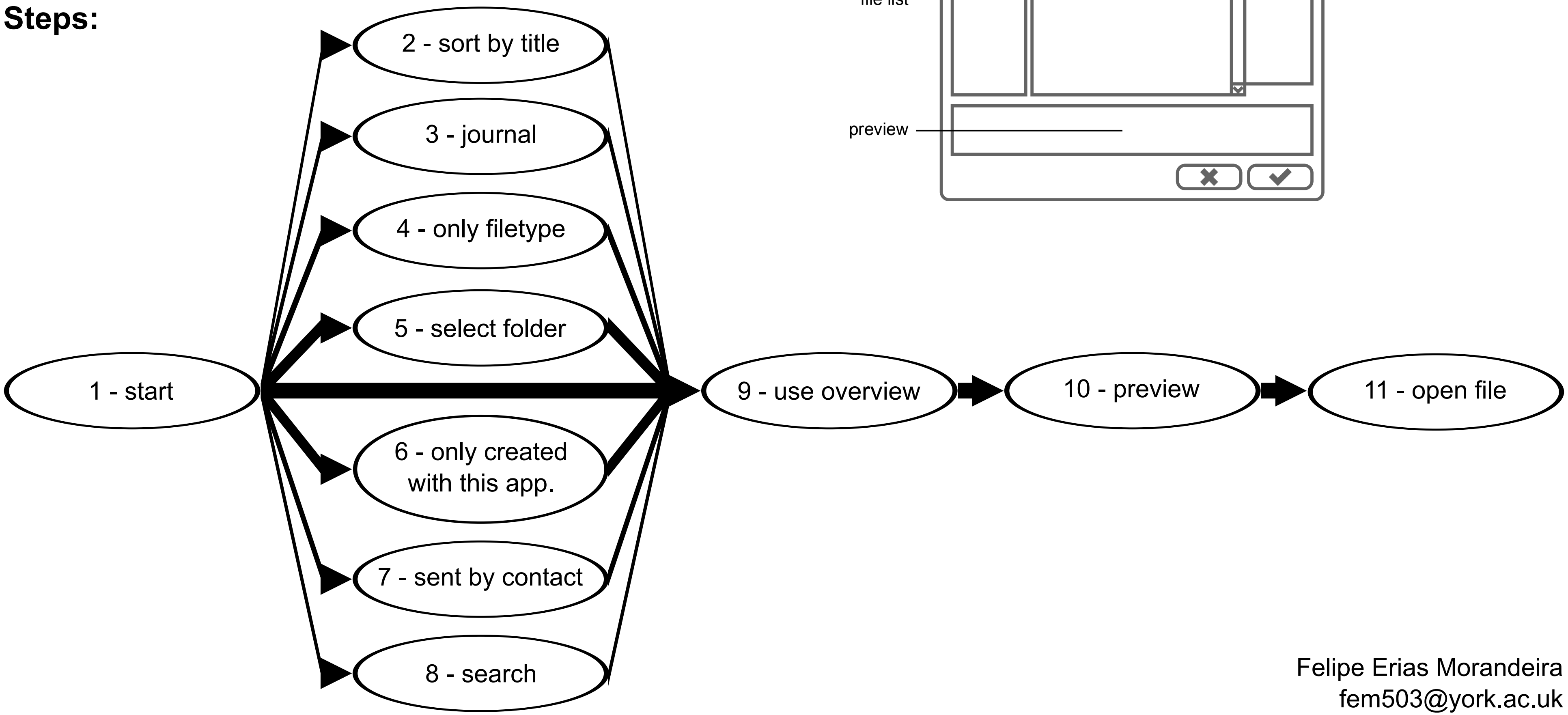
Cognitive walkthrough

Cognitive Walkthrough

a better way to open files from an application



Steps:

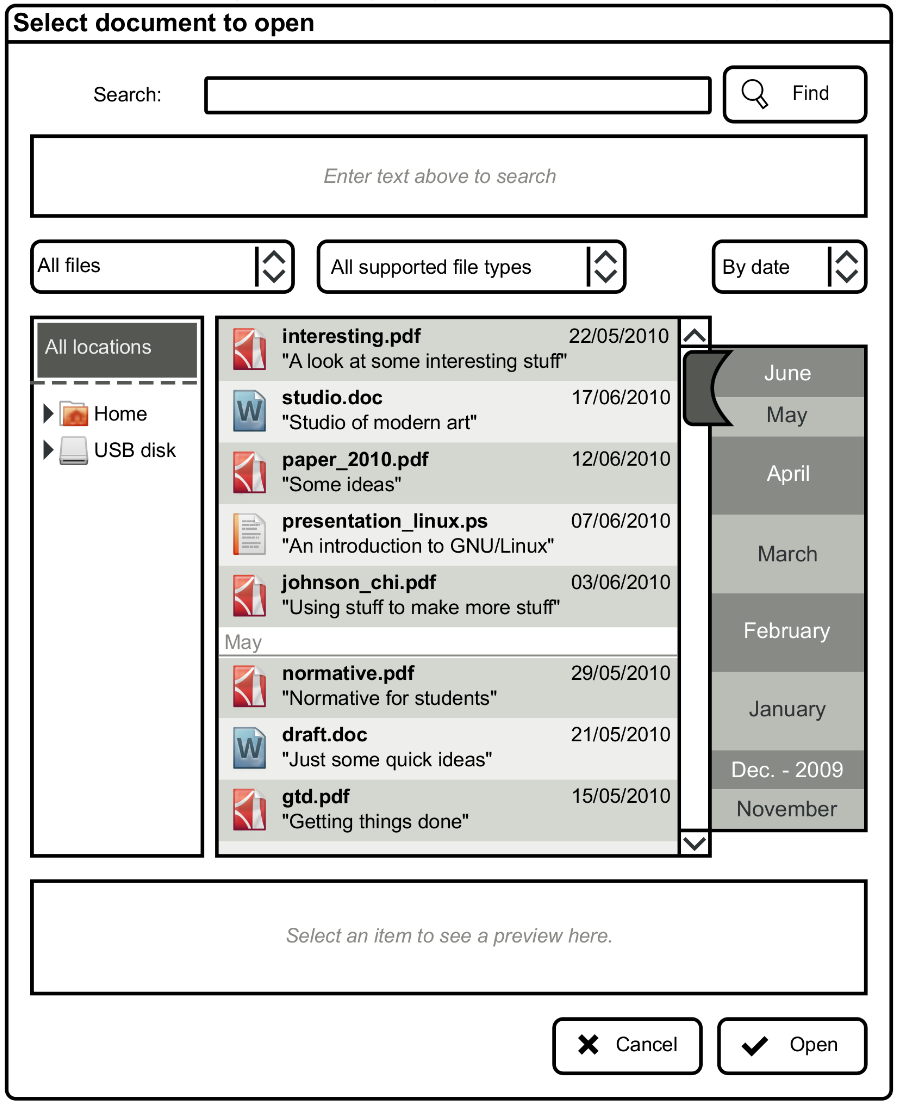


Felipe Erias Morandeira
fem503@york.ac.uk

Subtask: **1 - launch file chooser dialog**

Action: press the item File -> Open... in the application's menu, an "Open" button or similar

Result: the file chooser dialog shows up; by default all files are shown and they are ordered by date (in this case, creation/modification date)



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action?
(linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

4a. Comments as to why/why not:

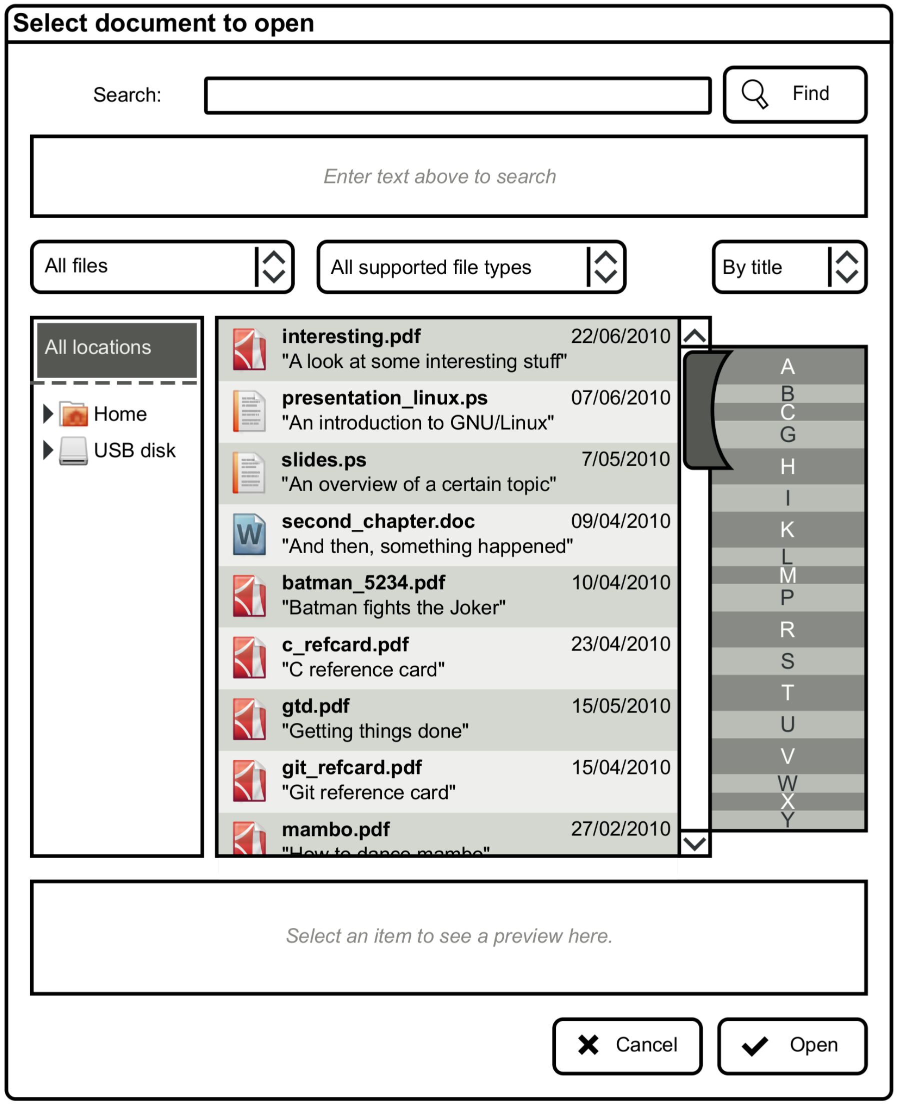
4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

0 25% 50% 75% 100%

Subtask: **2 - sort by title**

Action: activate the combobox with the legend "By date" (default value) and select "By title"

Result: files are ordered by their title, the overview displays letters instead of dates



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action?
(linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

4a. Comments as to why/why not:

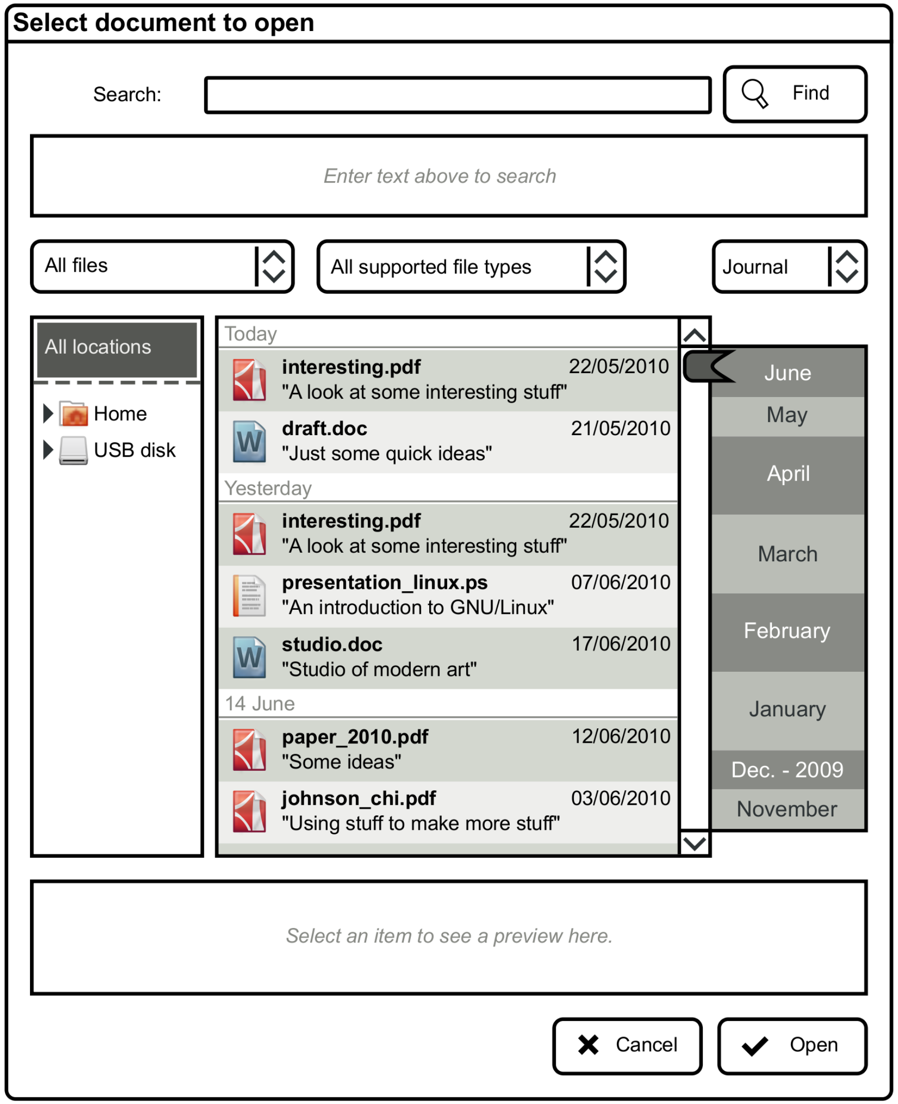
4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

0 25% 50% 75% 100%

Subtask: **3 - Journal**

Action: activate the combobox with the legend "By date" (default value) and select "Journal"

Result: the file list shows the files used in the past days; this makes the list longer because elements can appear several times in different days



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action?
(linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

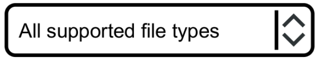
4a. Comments as to why/why not:

4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

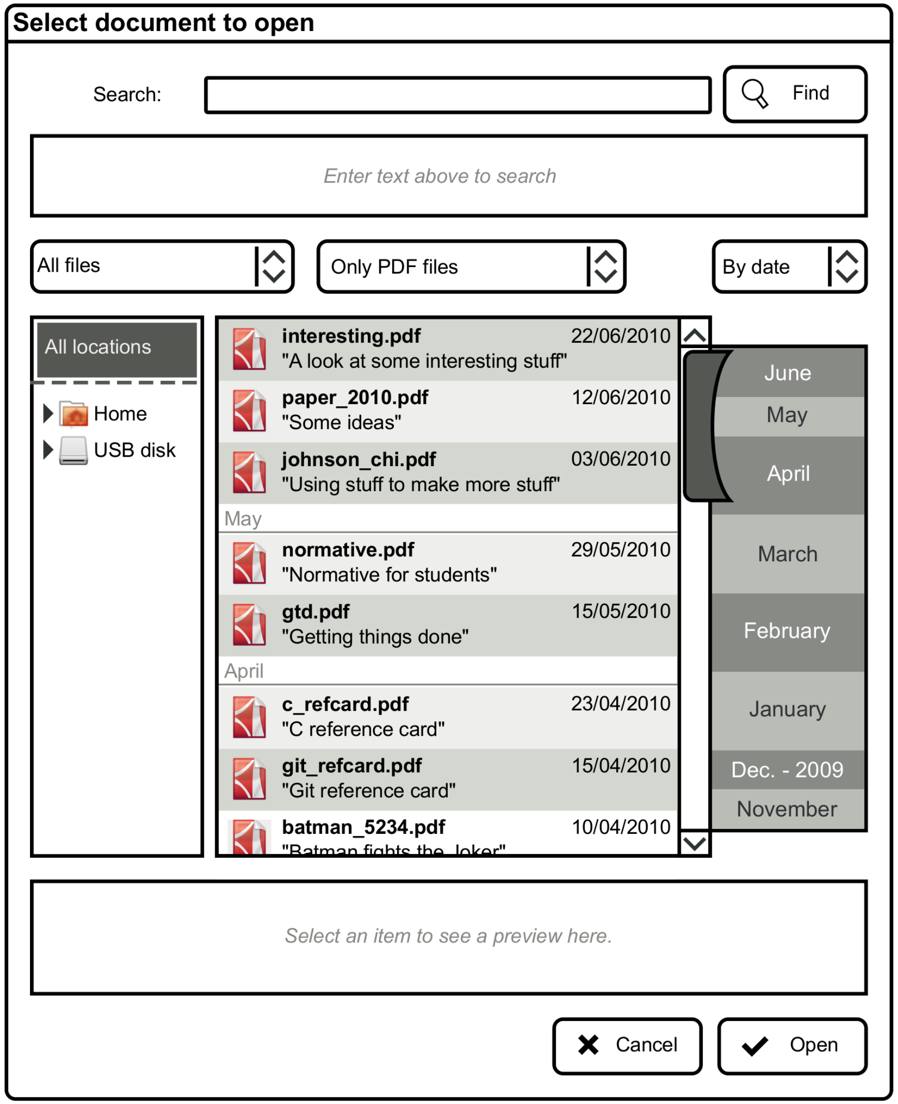
0 25% 50% 75% 100%

Subtask: **4 - only files of a certain type**

Action: activate the combobox with the legend "All supported file types" (default value) and select "Only PDF files"



Result: only PDF files are shown in the file list; the overview reflects this change



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action?
(linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

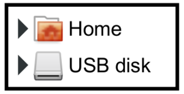
4a. Comments as to why/why not:

4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

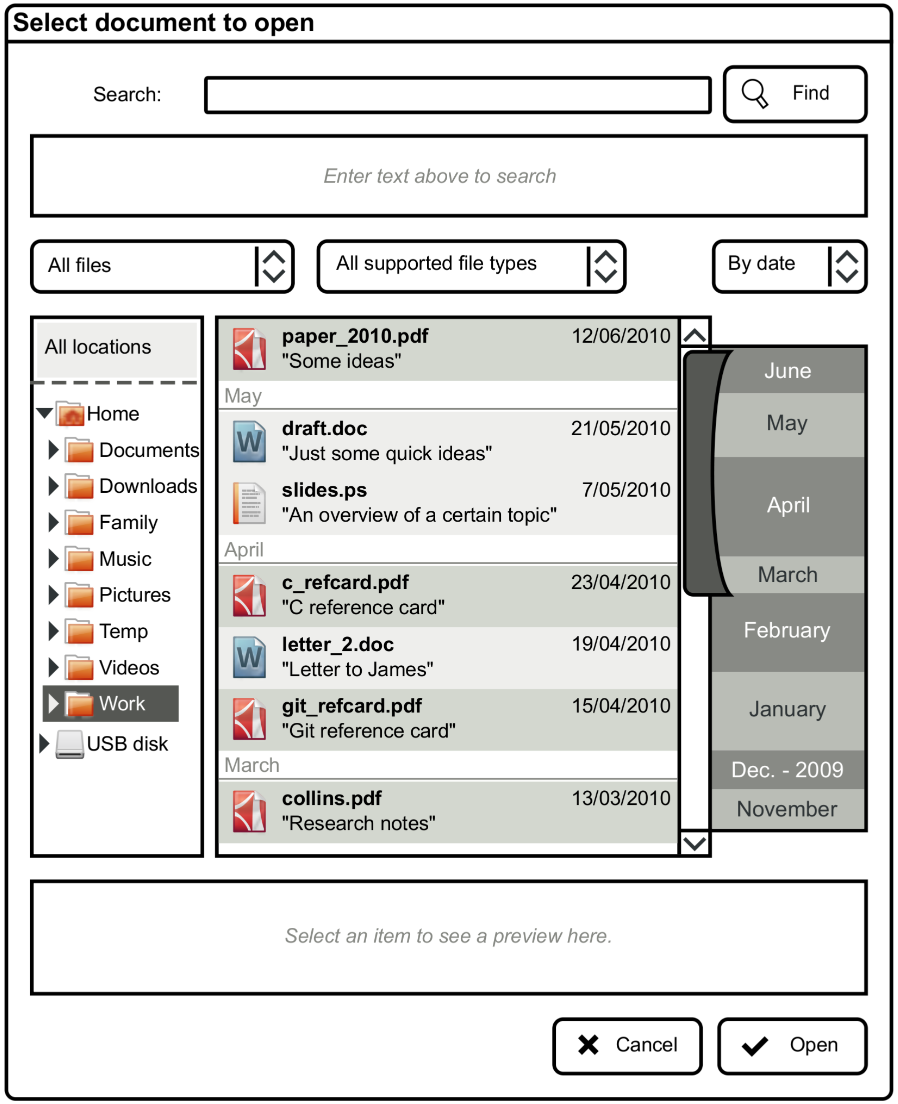
0 25% 50% 75% 100%

Subtask: **5 - select folder**

Action: use the folder browser on the left to explore the folder tree and select the desired folder



Result: the file list shows only the files contained in the selected folder



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action?
(linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

4a. Comments as to why/why not:

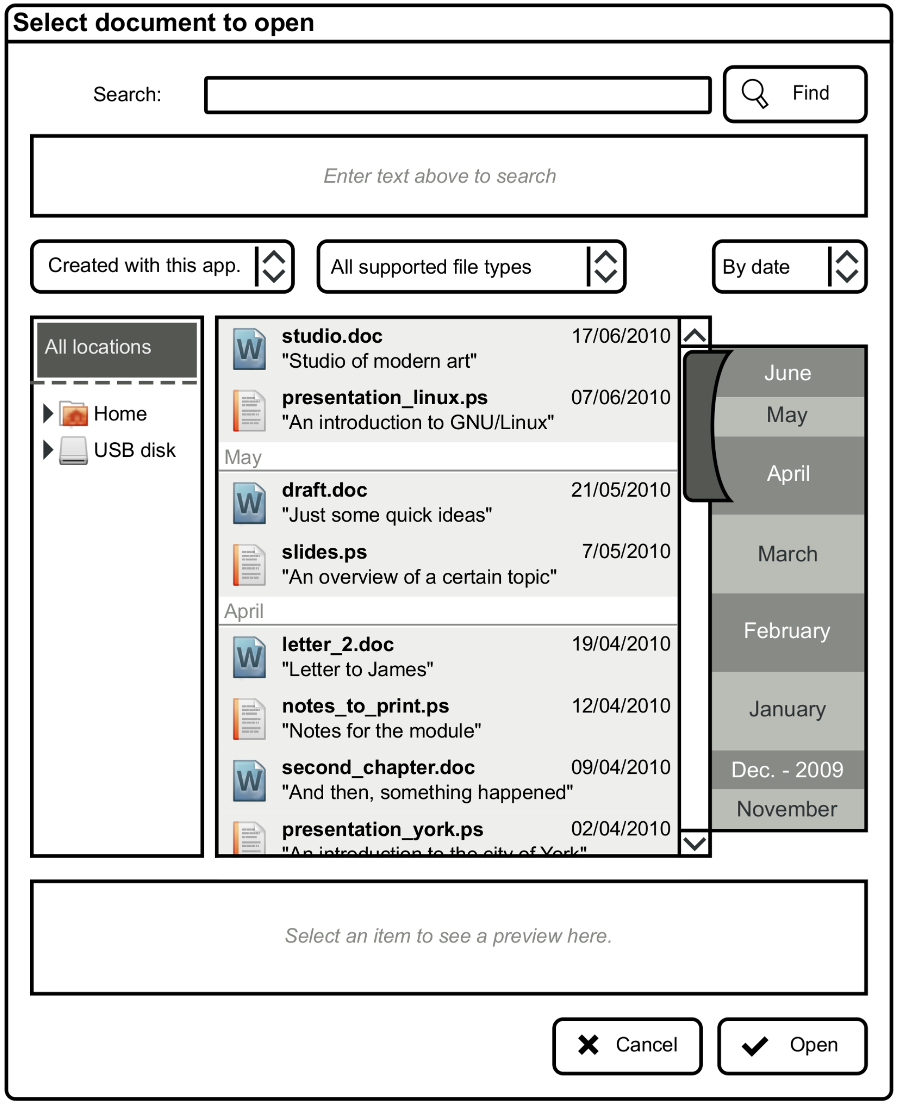
4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

0 25% 50% 75% 100%

Subtask: **6 - only files created with this app.**

Action: activate the combobox with the legend "All files" (default value) and select "Created with this app."

Result: only files that were created with the current application are shown in the file list



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action?
(linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

4a. Comments as to why/why not:

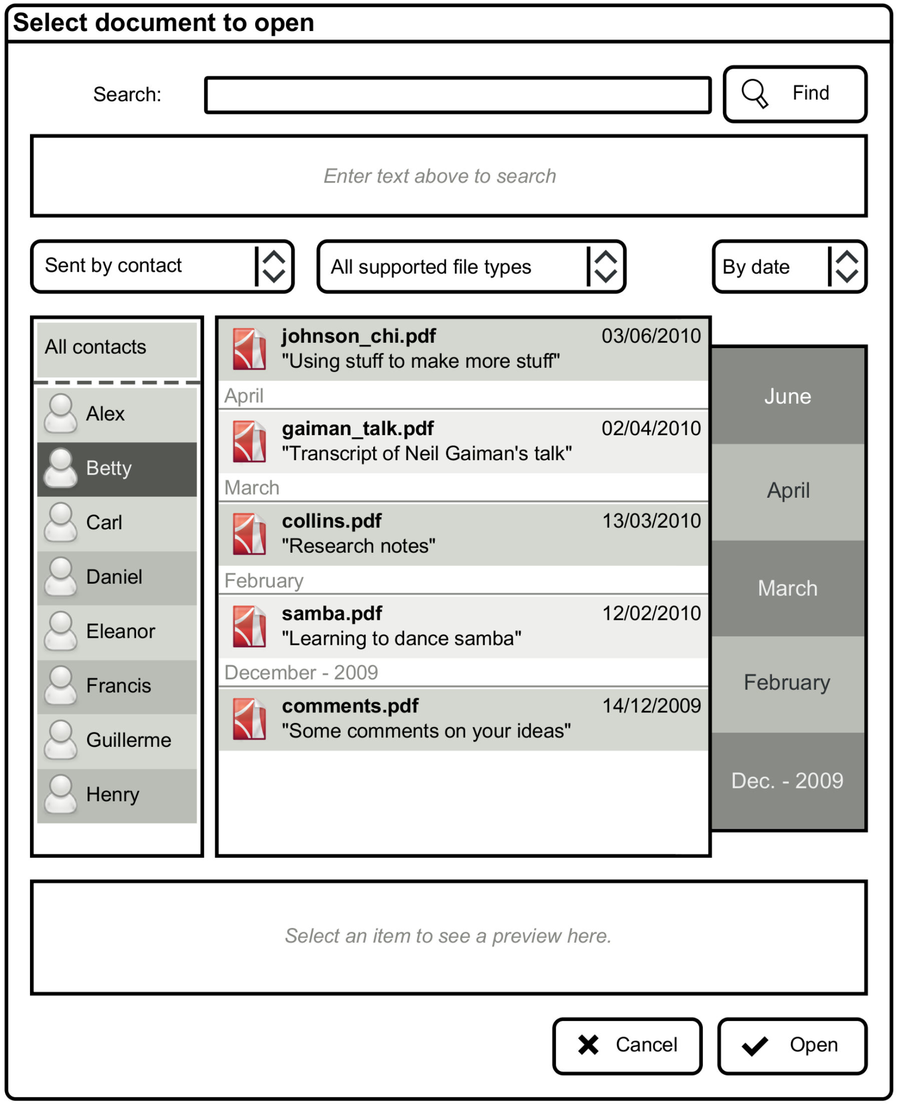
4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

0 25% 50% 75% 100%

Subtask: **7 - files sent by a contact**

Action: activate the combobox with the legend "All files" (default value) and select "Sent by contact"

Result: only files that were sent by a contact are are shown in the file list; the left selector shows a list of contacts to further refine the search (e.g. Betty)



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action? (linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

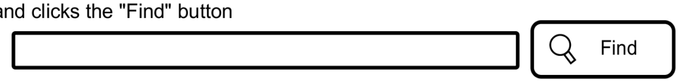
4a. Comments as to why/why not:

4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

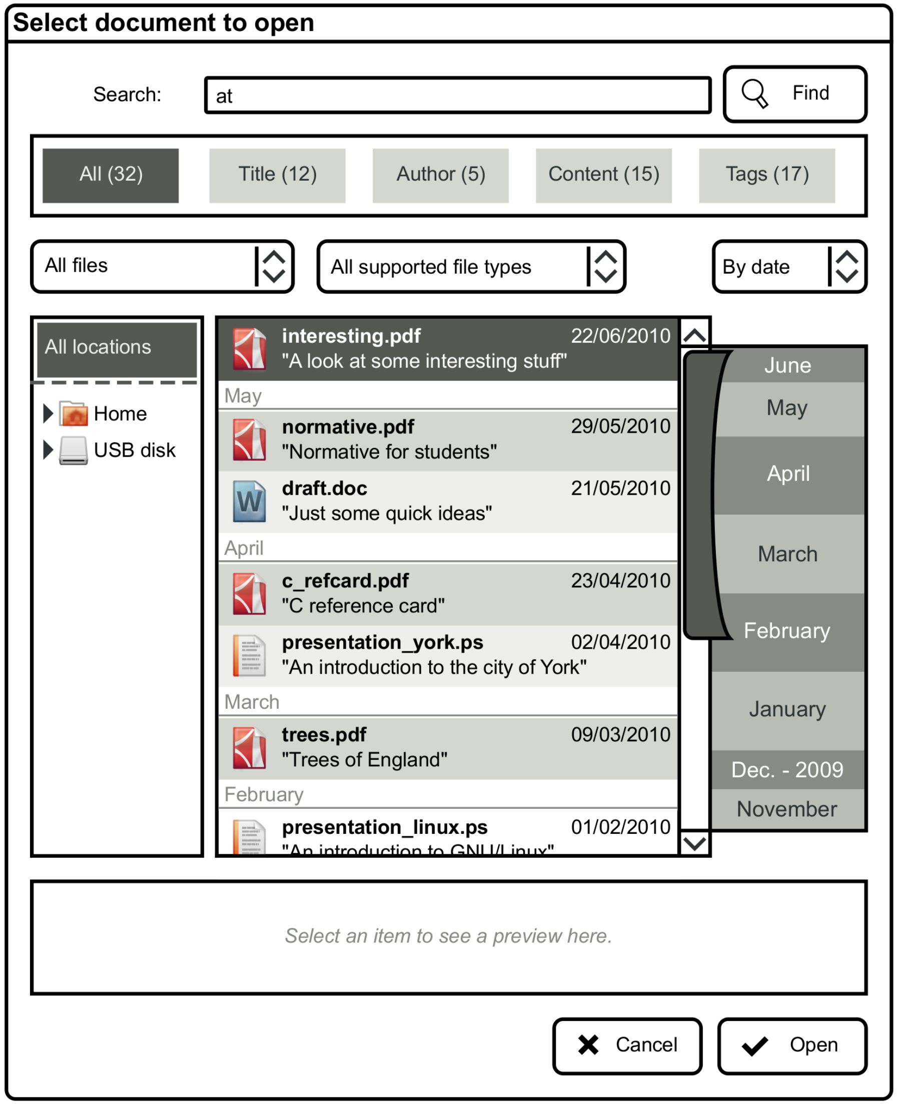
0 25% 50% 75% 100%

Subtask: **8 - search**

Action: the user enters one or more keywords in the text field and clicks the "Find" button



Result: the file list shows only the files that match the search
the user can refine the search by selecting only one of the properties to match



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action?
(linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

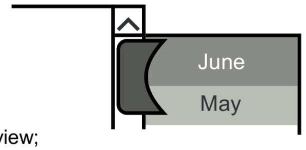
4a. Comments as to why/why not:

4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

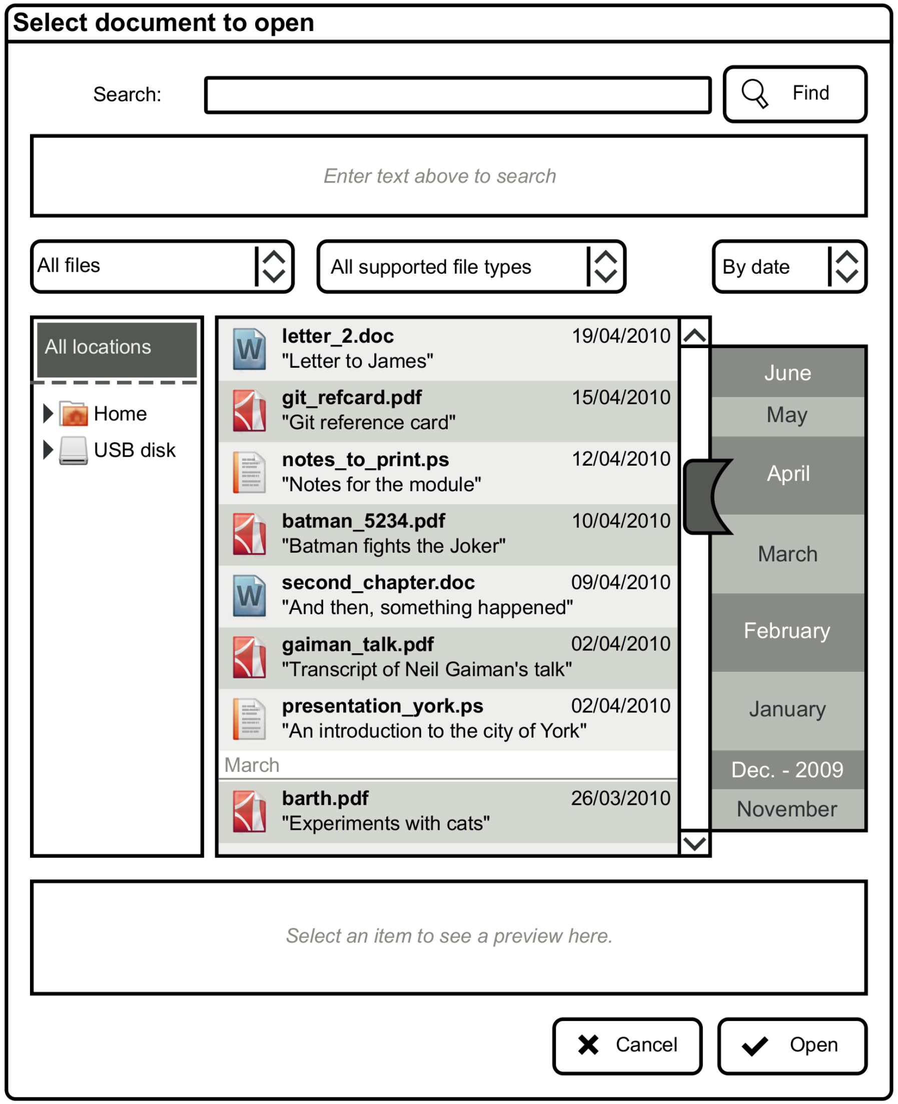
0 25% 50% 75% 100%

Subtask: **9 - use overview**

Action: at any time, the user can click in the overview to scroll the list to that point; this provides a fast way to manipulate the list



Result: the list scrolls to the point indicated by the overview; this subtask is to evaluate if this is a good way to navigate information



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action? (linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

4a. Comments as to why/why not:

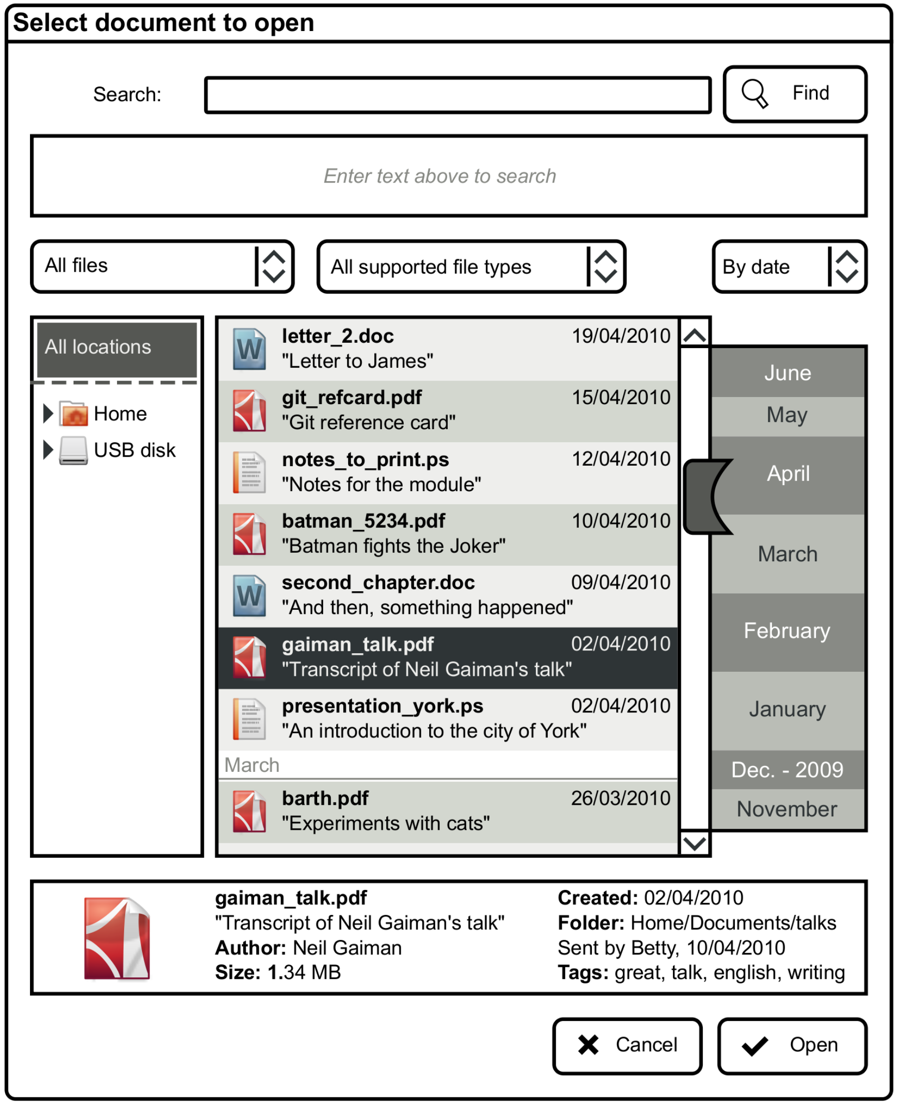
4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

0 25% 50% 75% 100%

Subtask: **10 - preview**

Action: the user clicks on one of the elements in the file list

Result: more complete information about the item is displayed in the area below; the intention is that the user can review his decision before clicking "Open"



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action? (linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

4a. Comments as to why/why not:

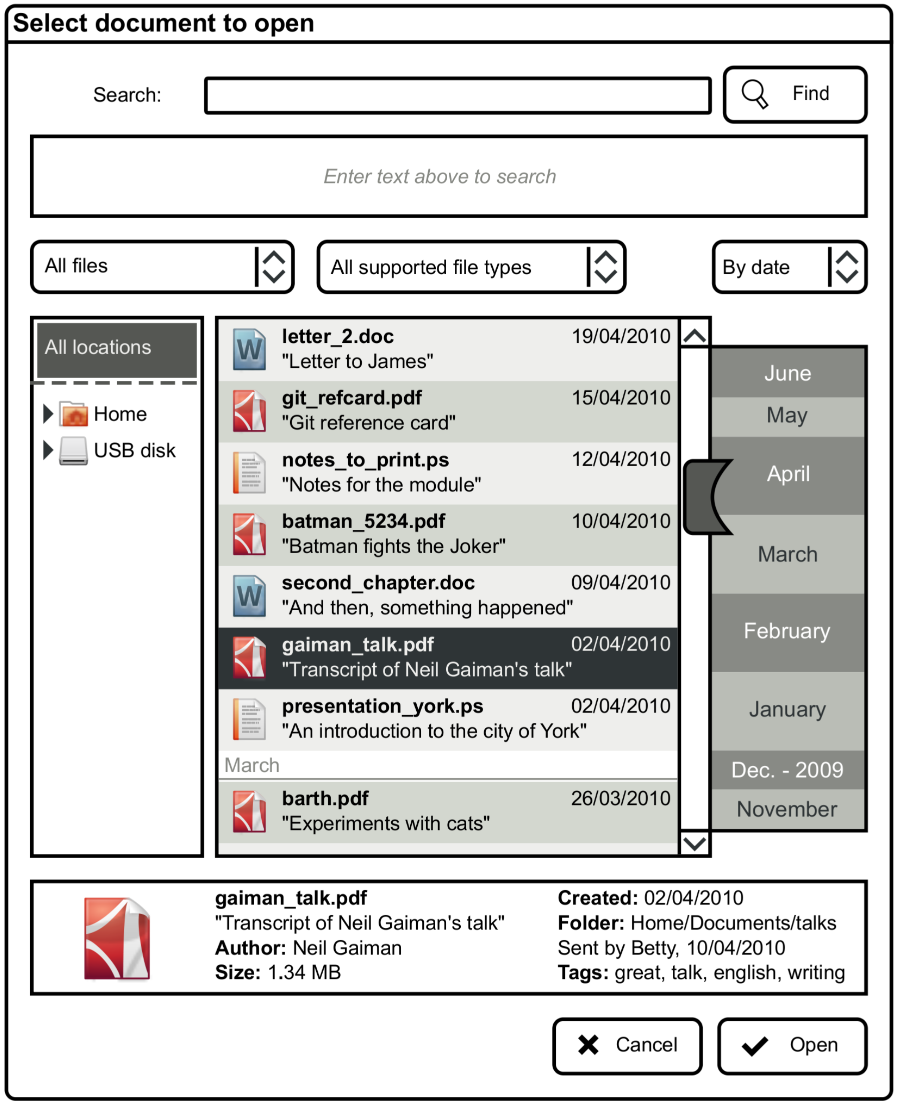
4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

0 25% 50% 75% 100%

Subtask: **11 - open file**

Action: The user clicks the "Open" button.

Result: The file is opened in the application that he was using.



1.Will the user understand that this step/subtask is needed to reach their goal?

Yes / No

1a. Comments as to why/why not:

1b. What is the likelihood they willl have a problem understanding the need for this step/subtask?

0 25% 50% 75% 100%

2.Will the user notice that the correct action is available?

Yes / No

2a. Comments as to why/why not:

2b What is the likelihood they willl have a problem noticing the the availability of the correct action?

0 25% 50% 75% 100%

3.Will the user understand that the required step/subtask can be achieved by the action?
(linking goal and action)

Yes / No

3a. Comments as to why/why not:

3b/ What is the likelihood they willl have a problem understanding the correct action?

0 25% 50% 75% 100%

4.Does the user get appropriate feedback if they make the appropriate action?

Yes / No

4a. Comments as to why/why not:

4b. What is the likelihood they willl have a problem noticing/understanding the feedback?

0 25% 50% 75% 100%

Appendix B

Experiment questionnaire

PARTICIPANT CONSENT FORM

Researcher: Felipe Erias Morandeira – fem503@york.ac.uk
Supervisor: Dr. Alistair Edwards – alistair@cs.york.ac.uk
Department of Computer Science, University of York, Heslington, York, YO10 5DD, UK

Experiment briefing

This experiment tries to validate the design of a dialog for opening files from a running application. For this, we will use a prototype of the user interface that operates on a synthetic list of files in order to provide a good idea of how the user interface would work without having to care about the specific internals of the tool. The experiment will consist on four tasks where you will be given a vague description of a file and will be asked to use the dialog to locate said file.

Consent statement

Your participation in this experiment is entirely voluntary; there will be no remuneration for the time you spend evaluating it. All data gathered from the usability study will be treated in a confidential fashion: It will be archived in a secure location and will be interpreted only for purposes of this evaluation. When your data are reported or described, all identifying information will be removed. There are no known risks to participation in this experiment, and you may withdraw at any point. Please feel free to ask the researcher if you have any other questions; otherwise, if you are willing to participate, please sign this consent form and proceed with the experiment.

Name:

Address:

Email:

Date:

Signature:

PRELIMINARY QUESTIONS

Gender: ☐ Male ☐ Female

Age:				
	☐ 0 – 15	☐ 25 – 34	☐ 45 – 54	☐ 65 – 74
	☐ 16 –24	☐ 35 – 44	☐ 55 – 64	☐ 75+

How often do you use computers?:

- ☐ never
- ☐ less than once a month
- ☐ between once a month and once a week
- ☐ every 2 – 3 days
- ☐ almost everyday

What desktop environment do you usually use?

☐ Windows XP	☐ Apple OSX
☐ Windows Vista	☐ GNU/Linux – GNOME
☐ Windows 7	☐ GNU/Linux – KDE
	☐ Other (please specify):

Are you aware that your computer has search facilities? ☐ Yes ☐ No

If yes, how often do you use them to find a document?

- ☐ less than once a month
- ☐ between once a month and once a week
- ☐ every 2 – 3 days
- ☐ almost everyday
- ☐ several times everyday

AFTER THE EXPERIMENT

Did you feel that the interface required a high level of mental effort?

Not at all 1 2 3 4 5 6 7 Very much so

Did you feel that the interface required a high level of physical effort?

Not at all 1 2 3 4 5 6 7 Very much so

Did you feel that the interface required a lot of time to use?

Not at all 1 2 3 4 5 6 7 Very much so

Did you feel that the interface let you successfully locate the files that you were looking for?

Not at all 1 2 3 4 5 6 7 Very much so

In general, did you feel that using the interface required a big effort?

Not at all 1 2 3 4 5 6 7 Very much so

Did you feel frustrated by the interface at any point?

Not at all 1 2 3 4 5 6 7 Very much so

How easy to use did you find this interface compared to the solution that you usually use?

Very easy 1 2 3 4 5 6 7 Very hard

How would you improve the interface? (Open question)

Bibliography

[1] A. Baddeley, *Your memory: a user's guide*. Carlton Books, 2004.

[2] D. Barreau and B. A. Nardi, "Finding and reminding: file organization from the desktop," *SIGCHI Bull.*, vol. 27, no. 3, pp. 39–43, 1995.

[3] V. Kaptelinin, "Creating computer-based work environments: an empirical study of macintosh users," in *SIGCPR '96: Proceedings of the 1996 ACM SIGCPR/SIGMIS conference on Computer personnel research*. New York, NY, USA: ACM, 1996, pp. 360–366.

[4] W. Jones, A. J. Phuwanartnurak, R. Gill, and H. Bruce, "Don't take my folders away!: organizing personal information to get ghings done," in *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2005, pp. 1505–1508.

[5] J. J. Kaye, J. Vertesi, S. Avery, A. Dafoe, S. David, L. Onaga, I. Rosero, and T. Pinch, "To have and to hold: exploring the personal archive," in *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*. New York, NY, USA: ACM, 2006, pp. 275–284.

[6] S. Dumais, E. Cutrell, J. Cadiz, G. Jancke, R. Sarin, and D. C. Robbins, "Stuff i've seen: a system for personal information retrieval and re-use," in *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. New York, NY, USA: ACM, 2003, pp. 72–79.

[7] M. Ringel, E. Cutrell, S. Dumais, and E. Horvitz, "Milestones in time: The value of landmarks in retrieving information from personal stores," in *Human–Conputer Interaction INTERACT '03*. IOS Press, 2003.

[8] J. Teevan, C. Alvarado, M. S. Ackerman, and D. R. Karger, "The perfect search engine is not enough: a study of orienteering behavior in directed search," in *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA: ACM, 2004, pp. 415–422.

[9] E. Cutrell, S. T. Dumais, and J. Teevan, "Searching to eliminate personal information management," *Commun. ACM*, vol. 49, no. 1, pp. 58–64, 2006.

[10] S. Cohen, C. Domshlak, and N. Zwerdling, "On ranking techniques for desktop search," *ACM Trans. Inf. Syst.*, vol. 26, no. 2, pp. 1–24, 2008.

[11] O. Bergman, R. Beyth-Marom, R. Nachmias, N. Gradovitch, and S. Whittaker, "Improved search engines and navigation preference in personal information management," *ACM Trans. Inf. Syst.*, vol. 26, no. 4, pp. 1–24, 2008.

[12] E. Adar, D. Karger, and L. A. Stein, "Haystack: per-user information environments," in *CIKM '99: Proceedings of the eighth international conference on Information and knowledge management*. New York, NY, USA: ACM, 1999, pp. 413–422.

[13] D. R. Karger and D. Quan, "Haystack: a user interface for creating, browsing, and organizing arbitrary semistructured information," in *CHI '04: CHI '04 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2004, pp. 777–778.

[14] R. S. Thorsten Prante, Kostanija Petrovic, "Kontextbasiertes individuelles informationsmanagement fuer das ubiquitous computing," in *Proceedings of Mensch & Computer 2004 (MC '04)*, 2004, pp. 291–300.

[15] T. Prante, J. Sauer, S. Lotfy, and A. Schmidt, "Personal experience trace: Orienting oneself in one's activities and experiences," in *CHI '10 Workshop – Know Thyself: Monitoring and Reflecting on Facets of One's Life*, 2010.

[16] B. A. Nardi, "Concepts of cognition and consciousness: four voices," *SIGDOC Asterisk J. Comput. Doc.*, vol. 22, no. 1, pp. 31–48, 1998.

[17] G. team, "Zeitgeist - gnome live!" http://live.gnome.org/Zeitgeist.

[18] D. Baur, "Visualizing media and music histories," in *CHI '10 Workshop – Know Thyself: Monitoring and Reflecting on Facets of One's Life*, 2010.

[19] L. Ltd., "Last.fm," http://www.last.fm.

[20] ——, "Build last.fm: Extend your last.fm experience," http://build.last.fm.

[21] B. Moore, M. V. Kleek, D. R. Karger, and M. Schraefel, "Assisted self reflection: Combining lifetracking, sensemaking and personal information management," in *CHI '10 Workshop – Know Thyself: Monitoring and Reflecting on Facets of One's Life*, 2010.

[22] Y. Inc., "Yahoo! directory," http://dir.yahoo.com/.

[23] G. Inc., "Google," http://www.google.com/.

[24] C. Marlow, M. Naaman, D. Boyd, and M. Davis, "Ht06, tagging paper, taxonomy, flickr, academic article, to read," in *HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia*. New York, NY, USA: ACM, 2006, pp. 31–40.

[25] M. Weiland and R. Dachselt, “Facet folders: flexible filter hierarchies with faceted metadata,” in *CHI '08: CHI '08 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2008, pp. 3735–3740.

[26] A. Cooper, *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity*. Pearson Higher Education, 2004.

[27] Google, “Google instant,” http://www.google.com/instant/#utm_campaign=launch&utm_medium=van&utm_source=instant.

[28] H. Petrie, “Cognitive walkthrough (short form) reporting form,” http://www-course.cs.york.ac.uk/rmh/Short%20Form%20Cognitive%20Walkthrough.doc.

[29] NASA, “Nasa tlx: Task load index,” http://humansystems.arc.nasa.gov/groups/TLX/.