Supporting Integration of Handwritten Information with Established Digital Tools

Master Thesis

Pierre-Olivier Trabichet
<po@student.ethz.ch>

Prof. Dr. Moira C. Norrie
Adriana Ispas
Michael Nebeling

Global Information Systems Group
Institute of Information Systems
Department of Computer Science

21st August 2011
Abstract

We present Glopim, a platform allowing the integration of handwritten information with established digital tools. This system aims to combine the simplicity and the flexibility of paper-based notetaking with the strengths of digital storage, while avoiding the hassle of the post-capture transcription phase.

The Glopim platform comes with an enhanced paper notebook allowing the classification of notes directly on paper, the possibility for the users to define their own set of metadata and a flexible architecture supporting the development and integration of output interfaces. Further, an output interface for WordPress, which was chosen for its growing popularity and the customisation opportunities provided by its architecture, is presented as proof-of-concept.

Glopim has been deployed and tested during three weeks within our research group here at ETH. A user study was conducted in order to evaluate the proposed pen and paper interface and to see how Glopim can improve the workflow of information workers. The obtained results show that Glopim is considered useful by the participants, in particular for digitalizing notes and sharing of sketches.
# Contents

1 Introduction 1

2 Related Work 5
   2.1 Tools for Personal Information Management 5
   2.2 Digital Pen and Paper Technology 9

3 Requirements Analysis and Workflow Definition 11

4 The Glopim Platform 15
   4.1 Personal Data Model 16
   4.2 Mapping with the Outputs 17
   4.3 Output Configuration 17
   4.4 The Paper to WordPress Workflow
      4.4.1 The Java Input 17
      4.4.2 The WordPress Output 18

5 Digital Pen and Paper Interface 19
   5.1 Capturing Information 19
   5.2 Send Signal 21
   5.3 Metadata Definition
      5.3.1 Using Tags 22
      5.3.2 Using Pre-Defined Buttons 22
      5.3.3 Using Custom Buttons 23

6 WordPress Interface 25
   6.1 The HTTP Interface Plugin 25
   6.2 The Enhanced Edit Panel 26
   6.3 The Notes Manager Plugin 27
   6.4 The Notes Manager Widget 28
   6.5 The Faceted View Plugin 28

7 User Study 31
   7.1 Study Design 31
   7.1.1 Participants 31
   7.1.2 Procedure 32
   7.2 Study Results
      7.2.1 Usage of Paper-Based Notes 33
      7.2.2 Glopim Usefulness 33
CONTENTS

7.2.3 Pen and Paper Interface Evaluation ........................................ 34
7.3 Concluding Remarks ............................................................. 36

8 Implementation and Extension Mechanisms ..................................... 37
  8.1 The Glopim Implementation .................................................... 37
    8.1.1 The Glopim Data Model ................................................ 38
    8.1.2 Inputs and Outputs Implementation ................................... 38
  8.2 Developed Extensions .......................................................... 40
    8.2.1 The HTTP Input ............................................................ 40
    8.2.2 The HTML Input ............................................................ 40
    8.2.3 The Email Input ........................................................... 41
    8.2.4 The AwareNews Output .................................................. 41
    8.2.5 The Social Outputs: Facebook and Twitter ......................... 43
    8.2.6 The Glopim Output ....................................................... 44

9 Conclusion & Future Work ........................................................ 45

A Glopim XML Files ................................................................. 47
  A.1 Personal Data Model File .................................................... 47
  A.2 Configuration File ............................................................ 48
  A.3 Favorite Metadata File ....................................................... 49

B Glopim Screenshots ............................................................... 51
  B.1 Welcome Page ................................................................. 51
  B.2 Java Input Page ............................................................... 52
  B.3 HTTP Input Page ............................................................. 53
  B.4 HTML Input Page ............................................................. 53

C Wordpress Screenshots .......................................................... 55
  C.1 Front End ................................................................. 55
  C.2 Notes Manager ............................................................... 56
  C.3 Edit Panel ................................................................. 57
  C.4 Faceted View ............................................................... 58

D Questionnaire ................................................................. 59
  D.1 General Questions ........................................................... 59
  D.2 The Glopim Platform - Part 1 ............................................ 60
  D.3 The Glopim Platform - Part 2 ............................................ 62
  D.4 Open Comments ............................................................. 62

E User Study Results .............................................................. 63
  E.1 Collected Data ............................................................... 63
  E.2 Likert Scale Questions ..................................................... 64
  E.3 Preference Questions ....................................................... 64

F Glopim Java Code ................................................................. 65
  F.1 Abstract Input Class ........................................................ 65
  F.2 Abstract Output Class ....................................................... 66
Introduction

Over the past few years, the growing ubiquity of Internet access and improvements made in software technologies have enabled the emergence of tools such as wikis, blogs or project management systems. These tools provide new ways to collect, store, retrieve, share and visualise data offering new opportunities for information workers. They also enable access to information from anywhere and support collaboration between people living across the globe. WordPress¹, an open-source content management system is a good example. What started in 2003 as a simple blog tool is now used by millions of people as a platform for storage and sharing of information. Moreover, its powerful plugin architecture allows developers to extend the platform in order to adapt it for different purposes.

Meanwhile, numerous information pieces are still generated out of these tools and slip out of their reach. Information workers write emails, use Post-its, scribble drawings on paper and take notes during meetings. Bernstein et al. [2] employ the term information scrap to speak about the information item that falls outside all PIM tools designed to manage it.

Paper-based notetaking especially is still very useful when it comes to capturing of information quickly and where specialised tools are not accessible. Studies showed the omnipresence of paper-based data in the personal information generated by a person and the central role it plays to them. Bernstein et al. [2] found during their experiments that Notebook and Post-it are two of the three most used tools to record information. The majority of engineers, for example, work with notebooks [17]. This can be explained by the intrinsic qualities of pen and paper: simplicity, quickness, and capability to generate drawings as easily as text. It is also proven that notetaking supports human memory. Kalnikaité et al. [12] demonstrated that taking notes aids people to focus on the incoming information and that these notes will later serve as triggers to help them remember information they would otherwise forget.

¹http://wordpress.org/
On the other hand, once the recording phase is done, paper information has multiple limitations: it is hard to index, difficult to share, and searching for relevant content through a whole notebook can be very time-consuming. Digital information does not have these disadvantages. Lin et al. [15] argue that paper is indeed more efficient for the record phase, but digital tools are better suited for the maintain and refer phases of the micronote lifecycle. Moreover, Kalnikaitė et al. [12] proved that paper-based notes utility degrades over time, unlike digital records which seem relatively immune to decay.

For these reasons, people work simultaneously with digital tools such as personal information management systems and more traditional techniques such as paper-based notetaking. This causes the generation of refined as well as unrefined data and leads to several problems such as information fragmentation [1], need for expensive post-capture transcription and difficulties to find and use relevant data. These problems highlight the need for solutions, such as the one we present in this thesis, that are able to automatically integrate paper-based information items with digital platforms. In this way, information workers can take advantage of the flexibility and simplicity of paper-based notetaking while benefiting from the strengths of digital storage.

Digital pen and paper technologies have been proposed as a solution to integrate paper-based information with the digital world. Various tools and software, such as the ones developed by Anoto², allow the interpretation of handwritten notes and their reproduction on digital formats. Although much work has been done in this area to provide means for browsing and reuse of handwritten notes on a computer, little has been done on the integration of such unrefined data with personal information management systems, especially web-based tools such as WordPress.

The goal of this thesis is to design and implement a platform allowing the integration of unrefined information with established digital tools. This platform is named Glopim for GLObal Personal Information Management and will be called as such in this text. As a proof-of-concept and a way to capture requirements, we focus on handwritten notes (as a source of unrefined data) and on WordPress (as an example of a web-based personal information management system). However, the Glopim platform has been designed in such a way that it can be easily extended and used with other tools and technologies.

This thesis is organised as follow:

- In the Related Work chapter, we analyse what has been done in the two areas of research related to this thesis: personal information management and digital pen and paper technology.

- In the Requirements Analysis and Workflow Definition chapter, we describe the approach taken to define the requirements of the Glopim platform and we present the workflow it has to support.

- In the The Glopim Platform chapter, we present the architecture and the functionalities of Glopim.

²http://www.anoto.com/
In the *Digital Pen and Paper Interface* chapter, we describe the pen and paper client used to capture handwritten notes as well as the way data and metadata are defined, stored and processed.

In the *WordPress Interface* chapter, we outline the plugins developed in order to enhance the functionalities of WordPress.

In the *User Study* chapter, we discuss the design, results and analysis of a user study conducted to evaluate the benefits for information workers consequent to the usage of the Glopim Platform.

In the *Implementation and Extension Mechanisms* chapter, we present the Glopim implementation and explain how our system can be extended to integrate new tools and technologies and in the meantime offer new possibilities to the user.

In the *Conclusion & Future Work* chapter, we give concluding remarks and provide suggestions for future work.
Related Work

In the previous chapter, we highlighted that information workers rely on digital systems such as personal information management tools but concurrently use paper-based notetaking. We also discussed the need for solutions that could reconcile these two paradigms. In this chapter, we present existing tools in the domain of digital notes and personal information management. Further, we provide examples of available digital pen and paper technologies.

2.1 Tools for Personal Information Management

In order to support notes management in the digital world, Van Kleek et al. [22] propose list.it: a lightweight browser-based tool developed to allow digital textual notetaking. This software was designed to be simple, fast, and easy to learn. A sidebar on the left of the window allows users to see their notes and to do searching by keywords. Using the capture bar on the bottom left, quick notetaking is always possible even when the sidebar is not open (see Figure 2.1). Simplicity and velocity appear to be the main advantages of list.it. However, it does suffer from some limitations. First, it only allows textual content and tends to be better for managing small notes rather than big ones. Second, directly sharing notes with fellow users is not possible. Finally, it only allows keyword searches and therefore metadata management is restricted. Despite this, Van Kleek et al. were surprised by the number of notes the users generated during the study they carried out. One third of the participants even continued to use this tool after the study ended indicating that they seemed to appreciate such systems.

Another slightly more elaborate solution is NotePals [5], a system developed to allow sharing of personal notes within a group of information workers. NotePals overcomes some of the limitations of list.it. First of all, the notes are stored in a shared repository accessible through any conventional web browser (see Figure 2.2), authorising easy sharing of notes with other users. NotePals also stores a set of basic metadata: author, subject, and creation time with each note. These metadata can later be used to do more specific searches and
facilitate the retrieving of relevant information. Moreover, the authors chose to rely on pen-based PDAs for the note recording phase. Their arguments are that PDAs can be used anywhere, support natural notetaking and can be easily synchronised with computers. On the other hand, recourse to PDAs tightly links NotePals with this technology and forces users to acquire and utilise such a tool. We can see the difficulties it may cause nowadays with touchscreen equipped smartphones being the new standard.
List.it and NotePals propose solutions to manage personal notes digitally. However, information workers usually possess numerous other digital items from different sources and in various formats and their is a need for applications helping them maintain, organise, retrieve and share all this personal information. MyLifeBits [9], Haystack [13], and HomeViews [8] are three personal information management tools proposed by their authors to realise these tasks.

MyLifeBits [9] is a SQL-based platform for personal information management. The authors’ idea is to provide the users with a surrogate memory, a tool saving and indexing everything they do: all accessed files are stored, every mouse and keyboard activity is recorded. It is also possible to define links between different pieces of information. A contact can, for example, be associated with a picture in which they appear. As shown in Figure 2.3, an interface is provided to access all these data. Through it, the users can retrieve relevant information items by querying the metadata and the links between data. The first drawback of MyLifeBits comes from the fact that everything is stored. Sellen et al. [19] argue that lifelogging is not the best solution because it generates too much information and does not support the natural way human memory works. A useful piece of information may, indeed, be drowned in the multitude of available ones. The second issue comes from the assumption that everything is (or at least will be) digital. For this reason, the integration of paper-based information has to be done through an extensive post-capture phase consisting of scanning the documents and manually adding the metadata.

For its part, Haystack [13] stores information using a semi-structured data model following the RDF [16] specifications. RDF represents information items as nodes and edges between them denote relations. An interesting point about Haystack is the fact that almost no data used by information workers is originally stored in a way compatible with RDF.
Hence, information has to be imported and converted from its original format. The authors provide a collection of extractors to do so. Documents in various formats, emails, or even instant messages can currently be integrated. One advantage of this method is that other extractors can be defined to integrate new types of data, supporting Haystack flexibility and extensibility. Haystack also comes with a user interface (see Figure 2.4) similar to information workers traditional tools in order to accelerate its learning and facilitate its use. One drawback of this system, especially when we compare it to web-based tools, is that sharing of information and collaboration is not directly supported.

Figure 2.4: A snapshot of the Haystack interface

HomeViews [8] is a personal data management system proposed by Geambasu et al. It differs from the two previous examples because it acts as a middleware, taking place between the file system and the application layer (see architecture in Figure 2.5). A novelty of this system is the possibility to access information stored remotely. For this purpose, HomeViews provides peer-to-peer [18] communication between different instances running on peer computers. It also specifies a capability-based access control schema allowing easy definition and resolution of access rights to the information belonging to fellow users. The authors also developed two applications working on top of their middleware: ViewGallery a photo organiser and ViewFS a simple tool allowing the creation and the management of directories on the file system. A criticism of HomeViews is the fact that it is not able to integrate existing applications and requires the development of brand new ones specifically shaped to its specifications.

All the tools presented in this section have a common drawback, they force users to use digital tools such as a computer or a PDA to record information. In certain cases, it would indeed be more convenient to rely on inconspicuous technologies, such as digital pen and paper, which would not change the notetaking workflow of the users.
2.2 Digital Pen and Paper Technology

Digital pen and paper technologies rely on enhanced pens designed to digitalise handwritten information items but still looking and behaving like regular pens. Therefore, they keep all the advantages of classic paper-based notetaking while offering new usage opportunities for the recorded data.

A good example of research done in this area is NiCEBook. NiCEBook [3] is a tool based on the Anoto\(^1\) technology and designed to enhance notetaking capabilities. It comes as a notebook (see Figure 2.6) and a digital pen. NiCEBook allows predefined categories such as **ToDo** or **Contact** to be assigned to notes directly on paper. An example of such an action is shown in Figure 2.7. User-defined topics can also be created and linked with notes using a similar process.

In addition to these enhanced paper features, NiCEBook also provides a software which notes taken in the physical notebook are synchronised with. This program offers a digital representation of the notes allowing actions, such as searching for keyword or visualisation of all the information items belonging to a specific category, which would not have been

\(^1\)http://www.anoto.com/
possible with a classical notebook. It also provides a simple interface to generate images from notes allowing users to send them by email for example.

NiCEBook demonstrates many of the advantages we can expect from a tool integrating paper-based notes into the digital world. However, it also comes with limitations. First, the notes have to be browsed in the tool provided by the authors forcing users to rely on it. Second, there is no easy way to share a note with other people. Finally, if someone wants to use a note in another program, the only available option is to export it as an image and reimport it somehow. This can be a problem if we, for example, want to integrate the textual content of a note in a document. In this case, the only solution would be to rewrite everything.

Txt-it [21] is a pen-and-paper based system proposed by Microsoft Research in order to allow the sending and receiving of text messages (SMS). This tool is designed for a very specific use, it therefore can not be compared to NiCEBook but still gives a good insight into the possibilities offered by digital pen and paper technologies. Txt-it is indeed able to interpret handwritten text, and send it as a SMS message through a GSM modem. An eventual response will be printed out allowing Txt-it to avoid the integration of any digital display. Thanks to this, the elderly, who are not always capable of using a mobile phone, can still use SMS to communicate.

Two points are particularly interesting about this project. First, the handwritten text is interpreted, showing that paper-based notes are not limited to image formats in the digital world and can also be integrated and reused as text. Second, the text is sent as a SMS message, proving that output for handwritten data is not restricted to a specific viewer provided with the digital pen.
3

Requirements Analysis and Workflow Definition

In the Related Work chapter, we saw that digital pen and paper technologies and personal information management solutions mostly come with new software supposed to resolve all the difficulties faced by information workers. Even HomeViews [8], which is primarily a middleware, only supports applications specifically designed for it. The problem of this approach comes from the fact that these new tools have to be accepted and learned by the users.

For this reason, we decided to tackle the problem from a different angle. Instead of providing new software, we chose to focus our efforts on integrating handwritten information with established tools. WordPress\(^1\), a platform which is already widespread through the community and used by many information workers, is used as proof-of-concept. Concerning the translation from handwritten notes to digital format, we opted for the Anoto\(^2\) digital pen and paper technology which has already been extensively used within our group at ETH. Furthermore, studies, like the one Brandl et al. [3] carried out to evaluate NiCEBook, showed that this technology, even if it is not perfect (handwritten recognition remains for example an issue), is mature enough to be used.

The first requirement of the Glopim Platform is to support the automatic integration of handwritten notes with the collection of digital data stored on the WordPress platform. This operation, in order to be as transparent as possible for the user, has to be done with minimal changes to the way notetaking is executed and data are accessed.

---

\(^1\)http://wordpress.org/
\(^2\)http://www.anoto.com/
Based on this first requirement, the easiest method to design our platform would have been to enhance WordPress in such a way that it can directly accept pen inputs. The drawback of this solution is that the platform would have been WordPress dependent and not usable with other outputs. For this reason, the next requirement of the Glopim platform is to be designed independently from the source and from the destination of information. It has to act as a bridge between the inputs and the outputs, being able to operate even if one of them is not available. Furthermore, we do not know which tools will be developed and used in the future. So, in order to guarantee the usefulness of the Glopim platform over time, extensibility has to be strongly supported. Therefore, another requirement is to provide a platform that can easily be enhanced with new output possibilities.

Consequently, the workflow of the Glopim platform can be defined as shown in Figure 3.1. This drawing served as basis for the design of the system presented in this thesis and will be enhanced in the next chapter where we present the Glopim platform architecture.

![Figure 3.1: The Glopim information workflow](image)

One of the major strengths of web platforms is the fact that they can be accessed from anywhere. Information workers only require a digital device and an Internet connection to retrieve their data. We want to provide the same opportunity to the users of our system. The next requirement of the Glopim platform is, therefore, to be accessible from anywhere through the web. Another quality of web-based platforms is the strong support they provide
for sharing information, and we saw that this functionality was missing in the majority of the tools presented in the previous chapter. To avoid this, the Glopim project has to be multi-user and to provide a way to send information items to other people. It should be possible to share information with the whole community or with certain people specifically.

Concerning the paper interface, NiCEBooK [3] showed that digital pen and paper technologies allow the enhancement of the notebook by providing actions that would not be possible with classical pens. The authors also demonstrated that users take advantage of these new opportunities to improve their workflows. Following the same idea, we want the Glopim platform to offer the possibility to its users to define metadata for their notes directly on paper. Furthermore, some metadata have special meanings depending on the output to which information is destined. WordPress, for example, uses tags to define keywords linked with an information item and categories to regroup items belonging to a similar topic. Therefore, the Glopim platform would have to provide a way to link the metadata written by the user on paper with these particular concepts. In the Related Work chapter, we also saw that handwritten notes can sometimes be useful as a drawing and sometimes as text. For this reason, another requirement of the Glopim platform is to be able to provide an image as well as an interpreted version of the notes to the different outputs.

On the WordPress side, one of the limitations of the basic platform is that it does not really support the creation of private information items. It is mainly blog post oriented. In order to provide new functionalities to WordPress and make it behave more like a classical personal information management system, the last requirement for this Glopim project is to provide a plugin allowing the easy management of personal notes on WordPress.

The requirements introduced in the previous paragraphs can be summarised as follow:

1. The Glopim platform has to provide a solution to automatically and transparently integrate handwritten information with established digital tools. WordPress will be used as proof-of-concept.

2. The Glopim platform has to be independent from the different input and output interfaces it contains and has to facilitate the development and integration of new ones.

3. The Glopim platform has to be available from anywhere through the web and has to enable sharing of information between users.

4. Paper-based notebooks have to be enhanced in order to offer the possibility to add metadata directly with the pens and the Glopim platform has to provide a way to map these user-defined concepts with the ones from the different outputs.

5. A WordPress plugin allowing creation and management of private information items has to be provided.
Based on the requirements and the workflow presented in the previous chapter, we propose the following architecture for the Glopim platform (see Figure 4.1). An incoming information item is received by the corresponding input. It is then interpreted based on the data model and the configuration specified by the user. Next, it is dispatched to the output channels where it will leave Glopim and will be transmitted further to the corresponding tools.

Figure 4.1: The Glopim architecture
4.1 Personal Data Model

Brandl et al. [3] discovered during their studies that all the participants used strategies to mark their notes. They also observed that, if people use different methods to do so, one person usually keeps the same system over time. Likewise, Kidd [14] noticed that information workers solve problems using structures internal to themselves and explained that it is better to propose tools to support this diversification than ones which try to standardise the way people work. For these reasons, we decided to provide a way for the users of the Glopim platform to define their own data model adapted to their preferences and personal work mode. These data models are used during the interpretation of information items arriving in the Glopim platform. In the prototype we present in this thesis, users personal data models are defined as XML files. One can imagine a metadata “priority”, which a user would use to set the importance of their notes. Possible values would be “low” or “high” and a specific information item should only have one priority. A token also has to be defined, it will be helpful to provide enhancements to the Glopim paper notebook (see Section 5.3.1). Such a “priority” concept could be defined by the user as follows:

```
1  <metadata unique="yes">
2    <name>priority</name>
3    <token>%</token>
4    <value>high</value>
5    <value>low</value>
6  </metadata>
```

An example of a complete data model file can be found in Appendix A.1. Users can modify their personal data models through a web browser. They have to log-in on the Glopim platform, download the current version of their XML file and upload it back when the desired changes are done (see Figure 4.2). More screenshots of the Glopim website are available in Appendix B.

![Figure 4.2: The Glopim web interface](http://www.w3.org/TR/2008/REC-xml-20081126/)
4.2 Mapping with the Outputs

The different outputs of the Glopim platform also come with their own concepts. WordPress, for example, uses tags and categories to classify the data, and each piece of information possesses a title. In order to represent these output-specific concepts, we decided to rely on a mapping file. The idea is that, if a user wants to use one of the concepts from WordPress, they have to map it to one of their own. As for the data model, this configuration has to be done in XML. If we want, for example, our “priority” metadata to be considered as a tag on WordPress, the following piece of code could be used:

```xml
< mappings>
    ...
    < mapping from="priority" to="wp.tag" />
</ mappings>
```

The wp prefix is here to denote a WordPress concept, user-defined concepts do not have prefixes. The mapping is part of a configuration file where other settings about the different outputs are also defined. An example of such a file can be found in Appendix A.2. Similar to the way users can adapt their data model, they can also modify their personal configuration file on the Glopim website.

4.3 Output Configuration

Output specific settings such as the url of the WordPress site have to be defined by the user in their configuration file we already introduced in the previous section (see Appendix A.2 for a complete example). In this file, the user can also define which outputs have to be used by default. It is possible to choose only one or to use a combination. However, Glopim also offers the possibility to override this default behaviour by using metadata. If someone, for example, sends an information item with “hello wordpress” as content and “wordpress” as value for a destination metadata, this note will be sent to WordPress and not to the outputs defined in the configuration file.

4.4 The Paper to WordPress Workflow

As shown in Figure 4.1, the current prototype of Glopim supports different workflows by providing several input and output interfaces. In this chapter and the two following ones, we focus on the transmission of information items from paper to WordPress. Other opportunities will be discussed in Chapter 8.

4.4.1 The Java Input

The digital pen and paper technology which we rely on for this project is based on Java. By providing a generic Java interface and not one specifically shaped for the digital pen, we allow any Java program to communicate with Glopim and increase, at the same time, the flexibility of the platform. Each time an information item is sent from the digital pen to Glopim, it is
received and decoded by this input before being transmitted to the WordPress output through the Glopim dispatcher.

### 4.4.2 The WordPress Output

While designing the WordPress output, the first difficulty we encountered was the fact that there was no easy way for us to send information items to it from Glopim. On the other hand, WordPress being open-source and fully customisable, we decided to create a plugin adding this missing functionality to our WordPress-based website. This plugin (as well as the other ones we developed) is described in detail in Chapter 6.

Once this limitation of the basic WordPress platform overcame, the implementation of this output was possible. First, each information item intended to be sent to WordPress is converted to a web form. This web form is then completed with the output-specific values such as the username and the password of the WordPress end-user. These output-specific details have to be defined by the user in their configuration file in order to be read automatically from there by the Glopim system. Once the form completion is over, it is sent to WordPress and interpreted by our plugin. Wordpress sends back the ID of the newly created item in its database. In the case of handwritten notes, a graphical representation of the notes is always attached to the interpreted content. This image is uploaded to the Wordpress platform in a second phase.

**The action metadata**

As explained in Section 4.2, output-specific concepts can be mapped to the user-defined metadata in a configuration file. In addition to tags, categories and titles used by WordPress to classify the information items it manages, we decided to define a fourth type: actions. This concept is used to represent the different ways information can be processed by WordPress. By default, an incoming piece of information is saved as a textual private note (only accessible to the user who wrote it). The actions are there to change this behaviour. A list of the available actions is shown in Figure 4.3. Actions are not unique and can be combined. It is, for example, possible to use “publish” and “image” at the same time. We can also send a copy to several users. However, it is important to be aware that the “publish” action overrides the “draft” action. Thus, if these 2 actions are used simultaneously, only the “publish” one will have an effect.

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>@image</td>
<td>The note will be displayed as an image and not as an interpreted text.</td>
</tr>
<tr>
<td>@draft</td>
<td>The note will be saved as a post draft and not as a private note.</td>
</tr>
<tr>
<td>@publish</td>
<td>The note will be directly published as a post.</td>
</tr>
<tr>
<td>@XXX</td>
<td>A copy of the private note is shared with the user XXX.</td>
</tr>
</tbody>
</table>

Figure 4.3: The WordPress actions
As part of the Glopim project, we developed a paper-digital interface using the Anoto\(^1\) digital pen and paper technology. The aim of this “client” application is to support the translation of handwritten notes into digital format and to send them to Glopim. This interface is based on the iPaper framework [10, 11], which can be installed on a computer in order to retrieve handwritten notes taken with a digital pen and store them in an object-oriented database. The architecture of our implementation of iPaper is shown in Figure 5.1

\(^1\)http://www.anoto.com/

5.1 Capturing Information

The paper interface of the system consists of a notebook, a metadata bookmark and a digital pen (see Figure 5.2). The pages of the notebook are covered with a non-intrusive dot pattern

---

Figure 5.1: Glopim iPaper Architecture
used by the pen to identify the coordinates of what is written. The transmission of digital data from the pen to the iPaper client can be done in two different ways:

- In the **online** mode, the pen establishes a bluetooth connection with the computer of the user and sends data as soon as it is available.

- In the **offline** mode, the pen stores everything in its internal memory. The data is transmitted later when the pen is put back into its docking station.

The main limitation of the online mode is the obligation to stay close to the computer. On the other hand, it also offers the possibility to give live feedback to users. In order to do so, we implemented a front-end which shows up on the lower-right of the computer screen (see Figure 5.3). This window displays information about the transferred data and about the Glopim platform status. The offline mode has to be used when the notetaking takes place somewhere else, e.g. during a meeting or during a journey. In this case, everything is stored in the pen and will be transmitted and interpreted by the computer in one operation when the pen is put back into its cradle.

![Figure 5.2: Glopim notebooks, metadata bookmark and pens](image)
5.2 Send Signal

In solutions such as NiCEBook [3] or CoScribe [20], the totality of the handwritten content is available on the computer. For the Glopim project, our approach is a bit different. If all the data is indeed digitalised in the client application, we do not want everything to be transmitted to the Glopim platform but only parts the user actively decided to send. In order to offer this flexibility, we defined a send signal in the form of a vertical line. Each time a user wants to send something to Glopim, they have to draw a vertical line on the paper and everything on the right of it will be transmitted (see Figure 5.4). The advantages of this solution are multiple. People can, for example, send only some parts of the information they wrote down on paper. They can also send information again with other metadata (see next section). On the other hand, one of the drawbacks is the fact that the line stays visible, introducing some noise in the notes on paper.
5.3 Metadata Definition

As mentioned in the requirements (see Chapter 3), we also wanted to enhance the notebook by offering classification options on paper. Thus, we decided to provide three methods to users for adding metadata to their notes directly with the pen.

5.3.1 Using Tags

During the evaluation of Note to Self, Van Kleek et al. [22] discovered that some of the notes contained extra terms which were not part of the content but added as metadata. Moreover, some of the users prefixed these terms with a sign in order to separate the different concepts. Based on these observations, we decided to provide a way to add metadata directly after the content of a note. Using the tokens defined in their personal data model (see Section 4.1), users can mark some terms as metadata and they will be interpreted as such by the system. An example is available on Figure 5.5.

![Figure 5.5: Adding metadata using tags](image)

5.3.2 Using Pre-Defined Buttons

Writing metadata directly after the content can in some cases present two major limitations. First, it relies on handwritten recognition and can therefore be misinterpreted. Second, the metadata has to be written down for each information item forcing the user to put it on paper again and again. In order to avoid these limitations, we developed a second method to add metadata in the form of a metadata card. This card is shaped like a bookmark and contains a list of buttons which can be clicked with the digital pen to add metadata to the notes before sending them (see Figure 5.6). An example of such a “metadata bookmark” can be seen in Figure 5.2. Moreover, users can define which metadata they want on their own bookmark accordingly to their own needs. This configuration can again be done in an XML file stored on the Glopim platform.

Unlike the metadata which are written down after the note content, the ones defined using buttons are persistent. This means that they will be attached to all the notes sent in the future. In this way, information workers can send a collection of notes with the same metadata context without having to redefine it all the time. A reset button is available on each page of the notebook to clear the associated metadata and go back to the default state.
5.3.3 Using Custom Buttons

One major limitation of current interactive paper solutions is the fact that they are fairly static. People have to write in specified areas and have to click in other dedicated areas for sending information or adding metadata. The send signal is already a way to offer more flexibility (mainly as it removes the static send button), but we also wanted to offer the possibility for users to define their own custom buttons. In this way, they can create buttons where they want, the size they want and with the content they want. This can be useful when a new metadata, which is not available on the metadata card, has to be used repeatedly.

We propose the following procedure to support the creation of buttons. First, the users have to write down the content of the future button on the paper. Second, they have to draw a rectangle around it (see Figure 5.7). These user-defined buttons, once created, work exactly like the bookmark ones (see Figure 5.8).

Figure 5.6: Adding metadata using the metadata bookmark

Figure 5.7: Creation of a custom button

Figure 5.8: Adding metadata using custom buttons
6

WordPress Interface

Once handwritten notes are collected using the pen and paper interface presented in the previous chapter, Glopim transmits them to WordPress where they can be accessed and managed. In order to improve the user experience of our system, we took advantage of the WordPress rich plugin architecture to develop three plugins enhancing our WordPress website.

6.1 The HTTP Interface Plugin

As explained in Section 4.4.2, a basic WordPress-based website allows the creation of data through a browser or using emails but does not offer an HTTP interface. The Glopim platform communication mechanisms being based on the HTTP protocol, we needed to find a way to provide this missing functionality to WordPress.

We developed a plugin for interpreting the HTTP requests sent by the Glopim platform and executing them on WordPress. This plugin defines a page which serves as an interface and processes all the requests arriving on it. Provided that the username and password are correct, the content of the request is used to create a new entry in the WordPress database. All the WordPress-based concepts (tags, categories, and title) are properly translated and the actions (see Section 4.4.2) executed. The remaining metadata (the ones defined by the user but not mapped to WordPress concepts) are saved as custom fields. The HTTP interface plugin is also responsible for uploading an eventual image, such as a handwritten drawing, and for linking it with the proper information item. When all these operations are done, an HTTP answer containing feedback is transmitted to the Glopim platform. In this way, the system can know if an error occurred.

1http://codex.wordpress.org/Custom_Fields
6.2 The Enhanced Edit Panel

Thanks to the HTTP Interface plugin presented in the previous section, handwritten notes can now be integrated in WordPress. In order for information items to be edited, WordPress provides an edit panel where the title, the content, the tags and the categories can be modified. However, this default edit page is too generic and therefore we decided to provide other functionalities to it. The screenshot of the enhanced edit panel can be seen in Appendix C.3. Here, we are going to present the two main improvements we made.

The Image Box

By default any handwritten piece of information arriving on WordPress is interpreted and stored in a textual format. However, the handwriting recogniser is not 100% reliable and interpretation mistakes can occur. It is also possible that a user actually wants to use the image and not the interpreted text. For these two reasons, we decided to provide an additional box containing the image (see Figure 6.1). Thanks to this new box, users can check if the interpretation of the text was done correctly. This box also contains an HTML “img” tag which can be used to integrate the image into the content.

![Figure 6.1: The image box](image)

The Custom Metadata Box

All the custom metadata defined by the user on the Glopim platform are also saved when the information item is stored in the WordPress database. Thus, it is important to provide a way to access and modify these metadata on the edit page. This is possible through our custom metadata box (see Figure 6.2). In this box, all the metadata type and value pairs are displayed and the users can delete or add new ones.
6.3 The Notes Manager Plugin

Another limitation of the basic WordPress platform is the lack of support for private and personal data. In order to address this, we decided to develop a plugin allowing the management of personal notes in the back-end of WordPress.

A new type of data, called “private note” was defined and we provide an interface to access and modify these notes. As shown in Figure 6.3, our notes manager plugin displays the notes in the form of small coloured post-its. The notes can be ordered alphabetically or by date of modification. It is also possible to display only notes in some colour. The plugin being developed with jQuery\(^2\) and AJAX, all the operations are done without reloading the pages making the usage more agreeable for the user. The notes are also automatically refreshed every 2 seconds. This means that newly arriving content, from the Glopim notebook for example, will be integrated without the need for users to manually reload the page.

The colour of the notes can be linked to a metadata in the settings page. Users have to choose which metadata have to be used and which value have to be linked with each colour. It is also possible to modify the colour of one specific note in the edit panel.

---

\(^2\)http://jquery.com/
6.4 The Notes Manager Widget

WordPress supports the development of widgets which can be added in the front-end of the platform. We decided to seize this opportunity to give Glopim users a direct link to their personal notes from any page of the Wordpress-based website. This widget (see Figure 6.4) appears as a post-it, present on every page of the website, and displays the number of notes of the current user. By clicking on it, users can access the notes manager page.

![Image of the notes manager widget](image)

Figure 6.4: The notes manager widget

6.5 The Faceted View Plugin

One of the main functionalities of the Glopim solution as a whole is the support of user-defined metadata. These custom concepts are maintained from the creation of handwritten notes in the Glopim notebook (see Chapter 5) to the storage of data in the WordPress database (see previous sections of this chapter). Even if these metadata can be accessed for each information item through the WordPress Edit Panel, there is a need for information workers to be able to browse their data and search for relevant content based on these concepts.

Phlat [4], a system proposed to optimise search of personal information, provides a filter area in its interface. This filter area allows users to specify values for different metadata in order to reduce the number of elements returned by a query. A similar method is presented by Yee et al. [23] to facilitate the browsing of large image collections and the user feedbacks are positive. 90% of them preferred the metadata approach to the keyword-based classical one. 97% of them also claimed that it helped them to get more insight about their data collection.

We decided, therefore, to develop a plugin allowing information workers to browse information items using a faceted view of the metadata. As we can see in Figure 6.5, each metadata concept is displayed on one line with toggle buttons for all possible values. The categories and tags are displayed in the two first rows followed by all the user-specific concepts sorted in alphabetical order. If no metadata is selected, all information items are displayed. If a user wants to narrow down the search, they can click on the provided buttons.
Another screenshot of the faceted view plugin is available in Appendix C.4.

As for the notes manager plugin, the combination of jQuery and Ajax allows us to avoid reloading pages. Each time metadata are selected or deselected, the list of results is dynamically updated. By clicking on one specific entry of the result set, the users can obtain an overview of its content. If they want to see more details or make changes, they can use the edit button. Deletion of an item is also possible by clicking on the respective button.

Figure 6.5: The faceted view plugin
In this chapter, we present a user study carried out to assess the benefits for information workers consequent to the usage of the Glopim Platform and to evaluate the proposed digital pen and paper interface. In the first section, we present the design of the study and, in the second one, we describe and analyse the quantitative results as well as the feedback from the users.

7.1 Study Design

The Glopim project having multiple facets, we decided to use a simplified test system for this user study in order to accelerate the learning of our tool and to focus on the aspects we wanted to assess. First, the same data model was specified for all the participants, and a unique metadata bookmark based on a default set of metadata was also created. This means that all the personalisation options of the Glopim platform as well as the existence of the Glopim website were hidden from the users. Moreover, the output possibilities were limited to WordPress and AwareNews [6] (see Section 8.2.4) in order to avoid the dispersion of data and make the analysis of the results easier. The rest of the test system was designed in accordance with the concepts presented in Chapters 4 to 6.

7.1.1 Participants

In order to obtain reliable results, we were looking for information workers who are dealing with a significant amount of information on a daily basis. We also needed to find people who were collaborating with each other using a WordPress-based system. Therefore, we recruited ten computer scientists who are members of the Global Information Systems Group here at ETH: seven of them currently doing a PhD and the three others writing their master thesis. These ten information workers work primarily on their computer, but also often participate in meetings, discussions and conferences. Further, WordPress is used within their group powering a web platform for sharing of information and discussion.


7.1.2 Procedure

Each participant received a digital pen with its docking station, two notebooks (one for the online mode and one for the offline mode) and two metadata bookmarks (again one for the online mode and one for the offline mode). The programs needed to collect the digitalised data recorded by the pen and to transfer them to Glopim were also installed on the participants computers. Further, an oral introduction of the system functionalities together with an explanatory document were also provided. In this way, users were aware of all the operations offered by the digital pen and paper interface, such as sending private messages to other or digitalising drawings. Finally, the participants were instructed to use the test system to generate private notes on the WordPress platform as well as to share information either in the form of a WordPress post or an AwareNews item.

After the participants had used the system for three weeks, all the transmitted data were collected and the users were asked to fill a questionnaire. This questionnaire, available in Appendix D, had multiple goals. First, it allowed us to get insight about how the participants normally manage their personal notes and their paper-based information. Second, it gave us the possibility to identify the benefits resulting from the use of the Glopim platform. Finally, it also permitted us to clearly measure the users satisfaction with the proposed digital pen and paper interface. This questionnaire contains 3 different types of questions. Some of them are based on a Likert scale; the answers of the users are rated between 1 (for the worst answer such as “strongly disagree” or “hardly ever”) and 5 (for the best answer such as “strongly agree” or “very often”), allowing us to obtain quantitative measures. Second, some of the questions ask the participants to choose the response they prefer among the proposed ones, enabling us to identify trends. Finally, a collection of open questions were also integrated in order to offer more answer flexibility to the user and capture unexpected suggestions.

7.2 Study Results

As explained in the previous section, we logged and collected all the information items created by the users during the three weeks of the user study. Further, we got their feedback through the questionnaire they had to fill in. It is important to note that only nine answered the questionnaire, the last one being on vacation when the study ended. A quick look to the data generated by the participants allowed us to notice that they mainly used the tool in an experimental way, testing the available functionalities. A total of 108 information items were transmitted through Glopim during the 3 weeks of the study. This means an average number of 11 items per user. The most active participant created 22 of them, when the less active one produced only 2. It is also interesting to see that the vast majority of the items were destined to WordPress, only 2 of them being shared on AwareNews. More detailed information about the collected data is available in Appendix E.1.

The collected information as well as the questionnaire results provided insight about the usefulness of the Glopim platform and the user satisfaction towards it. We are going to present our findings in the following subsections.
7.2.1 Usage of Paper-Based Notes

The first five questions of the questionnaire were designed to identify how often the users generate paper-based notes and how they manage them. Paper-based notetaking with a rating of 3.77 is the favourite notetaking and to-dos management technique of the study participants. One of them, for example, explained that he likes paper because it’s *always available, easier for free-form notes.*

The answers to question five, in which we asked the participant about how they manage their paper-based information, show that the behaviour of information workers towards their handwritten notes varies. While half of the participants take the time to transcribe and integrate part of them into digital tools, the other half keep everything on paper. Based on the limitations of paper-based information in terms of reuse and search possibilities, we assume this result is due to the fact that transcription tends to be time-consuming.

7.2.2 Glopip Usefulness

Based on their experiments with the prototype, the users were asked to evaluate the usefulness of such a system allowing the automatic integration of paper-based information with digital tools. We can see that the participants with an average rating of 4.1 (3 being “neutral”, 4 being “useful”, 5 being “very useful”) recognise the value of the proposed system.

Further, we wanted to identify the use cases where, according to the users, the Glopip platform could be helpful. In order to do so, we suggested six possibilities and asked the participants to give them a rating. As we can see in Figure 7.1, all the proposed use cases show positive trends. However, some of them got better results than others. It is the case of the “digitalise drawings and diagrams” one which got a 4.9 rating. Moreover, 20% of the 108 information items which passed through Glopip were tagged with the “image” action and therefore destined to be stored and used as images on WordPress. This can be explained by the fact that drawing still tends to be easier on paper than using a computer. For this reason,
the Glopim platform which is able to automatically transcribe them in digital format is very appreciated in this case. Based on the same idea, the transcription of notes in digital format is also highly rated. One user, for example, explained that notes of meetings or discussion with workmates are digitalised. Therefore I do not need a folder to keep such kind of paper notes. It is also interesting to see that “send information to others” is the lowest rated use cases. We assume this is due to the fact that WordPress is not usually used for private messaging. The development of an email output, for example, would probably increase this specific rating.

Another interesting fact which appeared during the study is that information items transmitted to the Glopim platform can be separated in two categories. The first one is composed of what we can call “post-it notes”. These notes are small (less than 50 characters) and contain mostly reminders or to-dos. Typical examples are Call Alice or Read paper about X. This category represents 80% of the information items created during the study and participants explained that they preferred the online mode to do so. We assume this is mainly due to the fact that they can get feedback from the Glopim frame. In this way, they can know which metadata are currently stated and if the Glopim platform was able to interpret correctly their data. The second category contains “meeting notes”. These notes, recorded during meetings or discussions with fellow workmates, are bigger, even sometimes distributed on several pages. They are usually generated out of the office, and therefore the offline mode is, this time, favored by the user.

7.2.3 Pen and Paper Interface Evaluation

The second half of the questionnaire was specifically designed to evaluate the pen and paper interface proposed by the current prototype of the Glopim platform.

The Send Signal

The current “vertical line” send signal seems to be appreciated by the participants. The average rating is 3.9 and only one of them rated it below 3. The main qualities they identified is that it is simple, clear and intuitive. One user, for example, said that it is an easy and intuitive way and handwritten notes normally have blank spaces around them which can be used. On the other hand, some participants complained about the fact that the lines are thereafter very notifiable on paper. One person also explained that it was sometime misinterpreted when doing drawings. One solution would be to use a slightly more complex signal like marking the corners of a note (see Figure 7.2). This signal would be less intrusive and would also offer the possibility to define exactly where the note ends. However, this requires two separate markers instead of one vertical line which may not always be appreciated.
CHAPTER 7. USER STUDY

Figure 7.2: Another variant of the send signal

Metadata Definition

We currently propose three different ways to add metadata to notes directly on paper and we were interested to get feedback from the users about them. As we can see in Figure 7.3, the metadata card is the method preferred by the participants. This method also gets the best rating (see Figure 7.4) and was used the most during the study, being responsible for 80 of the 94 metadata items added by the participants. Several reasons could explain this choice. First, this method tends to be fast and easy, the users having nothing to write down. Second, it does not suffer from handwriting recognition errors unlike the other two. Finally, combined with the feedback frame on the computer it is possible to know the current state of the metadata list.

However, this method limits the metadata choice to the list of options displayed on the bookmark. For this reason, some users preferred the custom buttons method, arguing that they can use their own metadata. The “annotating the content” method also has its supporters. These people like the facts that they can add metadata directly after the note and that they do not need to carry around the bookmark. It also tends to be convenient for specific metadata which will not be reused.

Figure 7.3: Preferred method  Figure 7.4: Rating of methods
The Handwriting Recognition Problem

The main criticism made by the participants concerned the handwriting interpretation. Indeed the handwriting recognition average rating is only 2.3 and the users very often had to correct the data once they were digitalised. One of the participants explained that *in my opinion the handwriting recognition should be improved first, because without a good recognition some features cannot be used properly*. The handwriting interpretation problems also undermine the metadata recognition and the creation of custom buttons. For example, we discovered by analysing the log files that people needed on average 5 or 6 trials to get a working custom button with the content they wanted.

On the Glopim side, there is not a lot we can do to change this as we rely on the handwriting recognition being provided by the pen and paper technology. However, the use of a slightly darker dot pattern on the paper, and the optimisation of the signals identification algorithms can enhance recognition results. It also seems likely that the recognition technology will continue to progress in the future as it did in the past few years.

7.3 Concluding Remarks

This study allowed us to confirm the hypothesis that Glopim, by its ability to automatically and transparently integrate handwritten notes with established digital tools, is useful to information workers. We were also able to identify digitalisation of information items and more particularly of drawings and diagrams as the bigger benefit for the users workflow. Furthermore, the proposed pen and paper interface was appreciated by the participants. The fact that the three proposed methods to add metadata received positive ratings (see Figure 7.4) and were preferred by some users (see Figure 7.3) is particularly interesting. It shows that these methods are complementary and support different notetaking processes. However, this user study also confirmed that handwriting recognition has to be improved in order for the Glopim platform to reach its full potential.
In this chapter, we present the implementation of the Glopim platform and explain the mechanisms available to support its extensibility. Further, we discuss additional input and output interfaces which were developed to offer new possibilities to the Glopim users.

8.1 The Glopim Implementation

The first design decision we had to make was which technology to use for the communication between Glopim and the different applications it receives input from and it sends output to. As our platform has to communicate with web-based tools and the transfer is always one way, free of any complex interactions, we decided to use HTTP\(^1\) requests. The advantages of HTTP are multiple: it is a well-known and well-documented networking protocol, it offers GET and POST methods exactly what we need, and it is simple to set up. We also chose to keep Glopim stateless (in accordance with the HTTP protocol) having no need for sessions because each arriving information item can be interpreted independently from others. Stateless systems are easier to design and simpler to understand. On the other hand, with every request starting a new transaction and the Glopim platform being multi-user, each arriving message will have to be properly authenticated.

The concrete implementation of the Glopim platform is based on Restlet\(^2\) a RESTful [7] web framework for Java. This framework offers abstractions for the HTML communication mechanisms, allowing us to concentrate on the Glopim system logic. Restlet also provides classes to implement HTTP basic access authentication, giving us an easy way to identify users and link them with their requests.

\(^1\)http://www.w3.org/Protocols/
\(^2\)http://www.restlet.org/
8.1.1 The Glopim Data Model

Even if the Glopim platform is working as a bridge and not storing any of the transiting information items, we need a representation for them during the time they spend in the system. An class diagram of the Glopim data model is shown in Figure 8.1.

Each piece of information processed by the Glopim platform is interpreted as a GObject (for Glopim Object). A GObject has a name, content, a list of GMetadata (used to represent the metadata associated with it) and an array of bytes (where a serialised image can be stored). This last attribute is for example useful to attach the graphical representation of a handwritten note. The GObject class also provides two methods responsible for the conversion of web forms\(^3\) to GOObjects and vice versa. These functions allow the Glopim platform to read and generate HTTP POST requests, supporting its communication mechanisms. A subclass (called GObjectIdentified) is also available, it adds a username attribute to a GObject allowing the linking of an information item with a user of the system.

The GMetadata object represents any metadata that can be added to a GObject. A GMetadata instance has a name and a value. The name could for example be “priority” and the value “high”. GMetadata also possess a token, which corresponds to the symbol defined by the user (see Section 4.1). The two last attributes (unique and possibleValues) are here to authorise some syntactical limitations. If the unique attribute is defined as true, it means that a GObject can only have one value for this specific metadata. A note with “high” and “low” priorities will in this case be rejected by the system. Similarly, the possibleValue attribute (if it is set) contains the list of authorised values. A note having a metadata with a value not in the list, will also be rejected by the system.

![Figure 8.1: The Glopim data model](image)

8.1.2 Inputs and Outputs Implementation

As we explained in Chapter 3, one of the requirements of the Glopim platform was to support extension in the form of new inputs or outputs. To do so, the majority of the Glopim logic was regrouped in two abstract classes: one for the input and one for the output. In this way,

\(^3\)http://www.w3.org/MarkUp/html-spec/html-specs.htmlSEC8.2.1
the design of a new interface is facilitated because the developers will only have to implement
the abstract methods of the corresponding abstract class, allowing them to focus on the code
specific to the new interface they are working on.

Inputs

The input interfaces of the Glopim platform work like web services. Using an HTTP GET
request, client applications can obtain details about a specific input, such as how to use it or
how to communicate with it. Using an HTTP POST request, they can submit information
items to the Glopim platform. Moreover, the Glopim platform will send a response for each
received HTTP POST request in order for the client to know if its request was correctly
interpreted by the system or not.

The abstract input class (see Appendix F.1) provides a method to read GObjects from
HTTP POST request. Thus to implement a concrete input class, the developers have to fill
the abstract method `represent()` with information about how their input interface works and
to provide a way for their client application to send HTTP requests. For the Java input (see
Section 4.4.1) for example, we provide a “GlopimClient.jar” which can be integrated in any
Java program in order to communicate with Glopim. Further, the `represent()` method gives
information about how to use this jar file.

Outputs

The architecture of the outputs is very similar to the one of the inputs. An abstract class
taking care of the majority of the logic is provided (see Appendix F.2), leaving only a few
specific details to be implemented in the concrete classes. A good example is the log out-
put which was developed for debugging and maintenance. This interface converts the cur-
rently processed GObject to a String and write it with all its associated metadata in the log
file. Figure 8.2 shows the source code of the log output class. All other Glopim operations
being defined in the GlopimAbstractOutput class, we only had to complete the `sendGObe-
ject(GObjectIdentified go)` and `getName()` methods.

```java
1 public class GlopimOutputLogger extends GlopimAbstractOutput {
2     private static final String OUTPUT_NAME = "logger";
3
4     @Override
5     protected String getName() {
6         return OUTPUT_NAME;
7     }
8
9     @Override
10     public void sendGObject(GObjectIdentified go) {
11         LOGGER.info(go.toString());
12     }
13 }
```

Figure 8.2: The Log output
8.2 Developed Extensions

In addition to the Java input and the WordPress output provided as proof-of-concept, we developed a few other interfaces to offer new opportunities to the Glopim users and to demonstrate the extensibility of the platform.

8.2.1 The HTTP Input

The Java input presented previously enables the communication between any Java-based application and Glopim, but we also wanted to offer a more generic input interface. For this reason, we decided to provide an HTTP input. To communicate with this interface, a client application has to emit an HTTP POST request in which the data are embedded in a web form. Explanations about the formats of the request and the form can be obtained by issuing a GET message. Through this input, any application (Java-based or not) can send information items to Glopim.

8.2.2 The HTML Input

We also wanted to provide a way to generate notes directly on the Glopim platform without having to use any specific input method. In order to do so, the Glopim platform automatically generates web pages containing an HTML form for each GObject defined by a user. An example of such a form is shown in Figure 8.3. The content of the notes can be written down in the content field and eventual metadata can also be added. Metadata, for which a list of possible values was defined by the user, appear in the form of a select list, the other ones in the form of a simple text field.

One limitation of the current implementation is the fact that only one value for each metadata can be added. In the next version, this could be revised in the form of a button allowing the addition of other occurrences of the not unique metadata.

![New mynote:](image)

Figure 8.3: An HTML form generated by Glopim
8.2.3 The Email Input

After notebooks, emails are the second source of information scraps identified by Bernstein et al. [2]. People strongly use email to communicate with each other or even to write reminders to themselves. Therefore, the integration of this type of information will be very beneficial to the Glopim users. The email input interface does not work exactly like the other ones. In place of waiting for HTTP requests, it actively checks every 2 seconds if new messages are available on a mail server through the IMAP protocol. Once a message is collected by the Glopim platform, it is parsed and converted to a GObject. From there, it is processed like any GOObjects generated by the other inputs.

The version of the email input delivered with the current prototype of the Glopim platform is limited. The messages have to be written following an exact structure to be correctly parsed by the system (see Figure 8.4 for an example), and there is no way to access eventual attached files or images. However, this should be revised in the next version where an enhanced parser could be provided.

![Image of an email interface](image)

Figure 8.4: Sending information to Glopim via email

8.2.4 The AwareNews Output

AwareNews [6] is a context-aware ambient news and awareness display invented here at ETH and used within our group. AwareNews regroups information from different sources and displays it on peripheral screens distributed around the building (see Figure 8.5). We decided to provide an AwareNews output in order for the members of our group to share notes and additional data directly using their pens.

\[\text{IMAP protocol} \]

AwareNews aggregates content from RSS feeds. Therefore providing an AwareNews output was equivalent to implementing a RSS one. For this reason, we developed a Java method which converts GObjects to RSS items. We offer the possibility to display the notes as text or image. If the users want one of their notes to be displayed as a drawing on AwareNews, our system stores the image and inserts an “img” tag into the RSS item (see Figure 8.6).

Figure 8.5: An AwareNews peripheral display

Figure 8.6: Handwritten content on AwareNews

http://www.rssboard.org/rss-specification
8.2.5 The Social Outputs: Facebook and Twitter

One of the biggest changes the society faced in the past few years is the emergence of social networking services. Facebook and Twitter are the two most used ones around the world with Facebook, for example, claiming more than 750 million active users\(^6\). For this reason we decided to develop Glopim outputs for the two platforms.

Twitter and Facebook supply an API allowing external developers to access their services. Moreover, Java libraries simplifying the use of these APIs are also available. Using Twitter4J\(^7\) and RestFB\(^8\), we were able to provide two interfaces between respectively Glopim and Twitter and Glopim and Facebook. Using these two outputs, Glopim users can share small notes on Twitter (see Figure 8.7) and publish notes and even images on Facebook (see Figure 8.8).

![Figure 8.7: Posting on Twitter through Glopim](image)


\(^7\)http://twitter4j.org/

\(^8\)http://restfb.com/
8.2. DEVELOPED EXTENSIONS

8.2.6 The Glopim Output

Glopim supports different workflows with its multiple input and output interfaces. However, flexibility can be further improved by allowing different Glopim instances to communicate with each other. In this way, several platforms can be channelled and tailored to different tasks. This new interface can, for example, allow Bob, who works for the organisation A, to send handwritten notes he took during a meeting to the Glopim platform deployed in organisation B. This second Glopim platform can be configured in such a way that everything coming from organisation A has to be sent as private notes to Alice in WordPress. In this way, Alice can access Bob’s notes using her WordPress account. (see schema in Figure 8.9).

The implementation of this Glopim Output was fairly easy. Indeed, since Glopim is based on Java, we used a GlopimClient to initialise a connection to the other Glopim platform. Moreover, since information items are already represented as GObjects, no conversion operations are needed.
Conclusion & Future Work

With the Glopim project, we present an implementation for automatically integrating handwritten information with existing digital tools. An enhanced paper notebook allowing information workers to classify and label notes directly with their pens is likewise proposed to extend the traditional paper-digital workflow. Glopim also offers the possibility to define a personal set of metadata for each user and guarantees the consistency of these concepts from paper to the destination. A WordPress output together with three plugins designed to enrich WordPress with support for personal information management are provided as proof-of-concept. Moreover, the architecture of Glopim has been designed to allow the extension of the platform for new output channels. Several examples of such add-ons are available in the current implementation, permitting, for example, the channelling of multiple Glopim platforms or the integration of handwritten data with Facebook and Twitter.

The user study conducted to evaluate the proposed tool shows that Glopim is found useful by the ten participants and improves information workers’ workflow. Its capacity to easily and transparently generate interpreted texts as well as images from handwritten data is particularly appreciated. Moreover, by proposing three different methods to add metadata directly on paper, Glopim is able to support various notetaking techniques and strategies.

Future work could extend the functionality of the current prototype. On the paper side, enhanced handwriting recognition could increase the reliability of the system. The possibility to cross things out on paper to delete the corresponding digital item is also an interesting idea. On the WordPress side, time-based reminders could be useful to help the users to respect eventual deadlines. A notification system for arriving private messages could also be convenient. On the Glopim side, we believe that the development of new output interfaces for other widely used tools such as Google Calendar or Evernote would increase the usefulness of the platform. In order to enrich further the support for extensibility of Glopim, the definition of input and output interfaces as independent plugins which can be added and removed when the system is running could also be tested.
A.1 Personal Data Model File

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<objects>
  <object>
    <name>mynote</name>
    <metadata unique="yes">
      <name>title</name>
      <token>$</token>
    </metadata>
    <metadata unique="no">
      <name>action</name>
      <token>@</token>
    </metadata>
    <metadata unique="no">
      <name>category</name>
      <token>*</token>
    </metadata>
    <metadata unique="no">
      <name>tag</name>
      <token>#</token>
    </metadata>
    <metadata unique="yes">
      <name>priority</name>
      <token>%</token>
      <value>high</value>
      <value>medium</value>
      <value>low</value>
    </metadata>
  </object>
</objects>
```
A.2 Configuration File

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<configs>
  <config ID="local" active="yes">
    <parameters>
      <mappings>
        <mapping from="action" to="receiver"/>
      </mappings>
    </parameters>
  </config>
  <config ID="catch" active="no">
    <parameters>
      <parameter name="url" value="https://api.catch.com/v1/notes"/>
      <parameter name="username" value="XXXX"/>
      <parameter name="password" value="XXXX"/>
    </parameters>
    <mappings>
    </mappings>
  </config>
  <config ID="facebook" active="no">
    <parameters>
      <parameter name="token" value="XXXX"/>
    </parameters>
    <mappings>
    </mappings>
  </config>
  <config ID="rss" active="no">
    <parameters>
    </parameters>
    <mappings>
      <mapping from="action" to="f_image"/>
      <mapping from="title" to="rss_title"/>
    </mappings>
  </config>
  <config ID="glopim" active="no">
    <parameters>
      <parameter name="url" value="XXXX"/>
      <parameter name="username" value="XXXX"/>
      <parameter name="password" value="XXXX"/>
    </parameters>
    <mappings>
    </mappings>
  </config>
  <config ID="logger" active="no">
    <parameters>
    </parameters>
    <mappings>
    </mappings>
  </config>
  <config ID="twitter" active="no">
    <parameters>
      <parameter name="consumerkey" value="XXXX"/>
      <parameter name="consumerscret" value="XXXX"/>
      <parameter name="accesstoken" value="XXXX"/>
      <parameter name="accesstroke" value="XXXX"/>
    </parameters>
    <mappings>
    </mappings>
  </config>
</configs>
```
APPENDIX A. GLOPIM XML FILES

A.3 Favorite Metadata File

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<favorite>
  <attribute>
    <content>@image</content>
  </attribute>
  <attribute>
    <content>@publish</content>
  </attribute>
  <attribute>
    <content>@note</content>
  </attribute>
  <attribute>
    <content>@pierre</content>
  </attribute>
  <attribute>
    <content>*fun</content>
  </attribute>
  <attribute>
    <content>*news</content>
  </attribute>
  <attribute>
    <content>#test</content>
  </attribute>
  <attribute>
    <content>%high</content>
  </attribute>
  <attribute>
    <content>%medium</content>
  </attribute>
</favorite>
```
B.1 Welcome Page
B.2 Java Input Page

Glopim 0.2
A robust bridge between java and wordpres

Java Interface

To call the Java interface from a Java program you have to follow these points:

- Add the GlopimClient jar to your project build path.
- Create a new GDBObject with the content you want to send.
- Create a new GlopimClient.
- Use the method "sendGDBObject" of your GlopimClient to send the GDBObject.
- The method "sendGDBObject" give a String back with the status of the request.

That’s it!

Here comes as simple example (for a GDBObject "myobject"):

```java
package glopim.example;

import java.util.List;

public class ClientExample {
    public static void main(String[] args) throws Exception {
        // Creation of an information item
        String informationItem = "Hello World";
        String name = "myobject";
        List<Metadata> metadata = null;
        GDBObject po = new GDBObject(name, informationItem, metadata);
        // Creation of a GlopimClient
        GlopimClient client = new GlopimClient("SRC", "USERNAME", "PASSWORD");
        // Sending
        String answer = client.sendGDBObject(po);
        System.out.println(answer);
    }
}
```
B.3 HTTP Input Page

HTTP Interface

To send a GObject to Glopim using the HTTP interface, you just have to send a
POST request to this page. The values stored in the POST request are dependent
of the object you want to send.

Concrete POST values for the GObject "mynote"

objectname = mynote
content = XXXXX
attr1name = title
attr1value = XXXXX
attr2name = action
attr2value = XXXXX
attr3name = category
attr3value = XXXXX
attr4name = tag
attr4value = XXXXX
attr5name = priority
attr5value = (high | medium | low)

B.4 HTML Input Page

New mynote:

<table>
<thead>
<tr>
<th>content</th>
<th>title</th>
<th>action</th>
<th>category</th>
<th>tag</th>
<th>priority</th>
<th>Submit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2011 All rights reserved
Farno Oliveri Trachsel
ETH Zurich
Wordpress Screenshots

C.1 Front End
C.2 Notes Manager

![Image of Notes Manager interface]

Notes Manager is a feature that allows users to organize and manage their notes. The interface is user-friendly, with options for sorting and filtering notes based on criteria such as date or title. It includes features for adding, updating, and deleting notes, as well as searching and categorizing them for easy access. Users can customize the interface and settings to suit their needs, making it a versatile tool for note-taking and organization.
C.3 Edit Panel

User Study has started

User study has started

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>External</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>1030</td>
<td></td>
</tr>
<tr>
<td>language</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
C.4 Faceted View
D.1 General Questions

1. When an idea (or something you have to do) occurs to you while you are sitting in front of your computer, how do you make sure to remember it?

<table>
<thead>
<tr>
<th>Option</th>
<th>Very often</th>
<th>Often</th>
<th>Sometimes</th>
<th>Rarely</th>
<th>Hardly ever</th>
</tr>
</thead>
<tbody>
<tr>
<td>I record it on paper.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I use an app on my smartphone.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I record it on my computer.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I rely on my memory.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other (please describe)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. When an idea (or something you have to do) occurs to you while you are not directly in front of your computer, how do you make sure to remember it?

<table>
<thead>
<tr>
<th>Option</th>
<th>Very often</th>
<th>Often</th>
<th>Sometimes</th>
<th>Rarely</th>
<th>Hardly ever</th>
</tr>
</thead>
<tbody>
<tr>
<td>I record it on paper.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I use an app on my smartphone.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I record it on my computer as soon as possible.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I rely on my memory.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other (please describe)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3. How do you manage your personal TODOs and Reminders?

<table>
<thead>
<tr>
<th>Option</th>
<th>Very often</th>
<th>Often</th>
<th>Sometimes</th>
<th>Rarely</th>
<th>Hardly ever</th>
</tr>
</thead>
<tbody>
<tr>
<td>I record them on paper.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I use an app on my smartphone.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I record them on my computer.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I rely on my memory.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other (please describe)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4. What is your preferred method and why?

5. How do you manage your paper-based notes?
   (a) I usually transcribe and integrate them into digital tools.
   (b) I sometimes transcribe important parts and integrate them into digital tools.
   (c) I manage them on paper and usually don’t transcribe them into digital tools.
   (d) I don’t have any paper-based notes.
   (e) Other (please describe)

D.2 The Glopim Platform - Part 1

6. What do you think about the possibility to automatically integrate paper-based information with digital tools?
   (a) Very Useful: It facilitates the management and the reuse of this information.
   (b) Useful: It can help in certain cases.
   (c) Neutral: It is equivalent to use paper and digital tools separately or to regroup them manually.
   (d) Useless: It is sometime worse than using paper and digital tools separately or to regroup them manually.
   (e) Very Useless: It is usually worse than using paper and digital tools separately or to regroup them manually.

7. Please assess the following statements:
   I think the Glopim platform is useful because it helps me to . . .

<table>
<thead>
<tr>
<th></th>
<th>Strongly agree</th>
<th>Somewhat agree</th>
<th>Neutral</th>
<th>Somewhat disagree</th>
<th>Strongly disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>. . . transcribe handwritten notes into a digital format.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . . manage my personal notes.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . . categorise information.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . . digitalise drawings and diagrams.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . . share notes with others (e. g. as a post).</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . . send information to others (e. g. as a personal message).</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. How do you think the Glopim tool could improve your personal work mode?
9. Which pen mode do you prefer for capturing information?

<table>
<thead>
<tr>
<th>Meeting notes</th>
<th>Online mode</th>
<th>Offline mode</th>
<th>Both</th>
<th>Not captured</th>
</tr>
</thead>
<tbody>
<tr>
<td>TODOs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reminders</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Message to others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other (please precise)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

10. How did you use your personal notes in WordPress?

<table>
<thead>
<tr>
<th>I accessed them using the plugin.</th>
<th>Very often</th>
<th>Often</th>
<th>Sometimes</th>
<th>Rarely</th>
<th>Hardly ever</th>
</tr>
</thead>
<tbody>
<tr>
<td>I managed them using the plugin.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I transformed them into posts.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I sent them to other users</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other (please describe)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

11. I like the proposed send signal (vertical line on the left of the text).

   (a) Strongly agree
   (b) Somewhat agree
   (c) Neutral
   (d) Somewhat disagree
   (e) Strongly disagree

12. Please explain your choice.

13. I like the following method to add metadata:

<table>
<thead>
<tr>
<th></th>
<th>Strongly agree</th>
<th>Somewhat agree</th>
<th>Neutral</th>
<th>Somewhat disagree</th>
<th>Strongly disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>...annotating the content</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...using the metadata card</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...using my own buttons</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

14. My preferred method to add metadata is ...

   (a) the one where the metadata are written directly after the content.
   (b) the one which relies on the metadata card.
   (c) the creation and usage of custom buttons.

15. Please explain your choice.
D.3 The Glopim Platform - Part 2

16. I am happy with the handwriting recognition.
   (a) Strongly agree
   (b) Somewhat agree
   (c) Neutral
   (d) Somewhat disagree
   (e) Strongly disagree

17. I had to correct handwritten information.
   (a) Very often
   (b) Often
   (c) Sometimes
   (d) Rarely
   (e) Hardly ever

18. I was able to create custom buttons as I wanted.
   (a) Strongly agree
   (b) Somewhat agree
   (c) Neutral
   (d) Somewhat disagree
   (e) Strongly disagree

19. Please explain your choice.

20. I was able to add the metadata I wanted.
   (a) Strongly agree
   (b) Somewhat agree
   (c) Neutral
   (d) Somewhat disagree
   (e) Strongly disagree

21. Please explain your choice.

D.4 Open Comments

22. Do you have any suggestions on how the current prototype could be improved?

23. Do you have any additional comments, observations or suggestions?
### User Study Results

#### E.1 Collected Data

<table>
<thead>
<tr>
<th>Data Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of participants</td>
<td>10</td>
</tr>
<tr>
<td>Number of items sent to Glopim</td>
<td>108</td>
</tr>
<tr>
<td>Average number of items per user</td>
<td>11</td>
</tr>
<tr>
<td>Biggest number of items per user</td>
<td>22</td>
</tr>
<tr>
<td>Smallest number of items per user</td>
<td>2</td>
</tr>
<tr>
<td>Number of items sent as image</td>
<td>24</td>
</tr>
<tr>
<td>Number of items sent as text</td>
<td>84</td>
</tr>
<tr>
<td>Number of big items (more than 50 characters)</td>
<td>23</td>
</tr>
<tr>
<td>Number of small items (50 characters or less)</td>
<td>61</td>
</tr>
<tr>
<td>Number of items sent to AwareNews</td>
<td>2</td>
</tr>
<tr>
<td>Number of items sent to WordPress</td>
<td>107</td>
</tr>
<tr>
<td>Number of items sent to both</td>
<td>2</td>
</tr>
<tr>
<td>Number of items directly published on WordPress</td>
<td>1</td>
</tr>
<tr>
<td>Number of items sent as draft on WordPress</td>
<td>4</td>
</tr>
<tr>
<td>Number of items sent as private notes on WordPress</td>
<td>102</td>
</tr>
<tr>
<td>Number of items sent to somebody else on WordPress</td>
<td>14</td>
</tr>
<tr>
<td>Number of metadata added using predefined buttons</td>
<td>80</td>
</tr>
<tr>
<td>Number of metadata added using custom buttons</td>
<td>11</td>
</tr>
<tr>
<td>Number of metadata added using tags after the content</td>
<td>3</td>
</tr>
</tbody>
</table>
### E.2 Likert Scale Questions

<table>
<thead>
<tr>
<th>Question</th>
<th>Average</th>
<th>Standard deviation</th>
<th>Min:</th>
<th>Q1:</th>
<th>Q2:</th>
<th>Q3:</th>
<th>Max:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>3.8</td>
<td>1.2</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>1.2</td>
<td>1.9</td>
<td>1.4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>1.3</td>
<td>3.4</td>
<td>0.9</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>1.4</td>
<td>2.7</td>
<td>1.1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>3.9</td>
<td>1.4</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>2.2</td>
<td>3.3</td>
<td>1.6</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>2.3</td>
<td>2.1</td>
<td>1.2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>2.4</td>
<td>3.0</td>
<td>0.9</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3.1</td>
<td>3.6</td>
<td>1.5</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>3.2</td>
<td>2.8</td>
<td>1.6</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>3.3</td>
<td>3.3</td>
<td>1.4</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>3.4</td>
<td>2.7</td>
<td>1.1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>3.1</td>
<td>1.4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>4.1</td>
<td>0.6</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7.1</td>
<td>4.2</td>
<td>0.7</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>7.2</td>
<td>3.6</td>
<td>0.7</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7.3</td>
<td>3.7</td>
<td>0.7</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7.4</td>
<td>4.9</td>
<td>0.3</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>7.5</td>
<td>3.6</td>
<td>0.7</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7.6</td>
<td>3.2</td>
<td>0.7</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>10.1</td>
<td>3.0</td>
<td>1.3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>10.2</td>
<td>2.2</td>
<td>1.1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>10.3</td>
<td>1.6</td>
<td>0.7</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>10.4</td>
<td>1.7</td>
<td>0.9</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>3.9</td>
<td>0.8</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>13.1</td>
<td>3.7</td>
<td>1.0</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>13.2</td>
<td>4.2</td>
<td>0.8</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>13.3</td>
<td>3.6</td>
<td>1.0</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>16</td>
<td>2.3</td>
<td>1.2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>4.4</td>
<td>0.7</td>
<td>3</td>
<td>4</td>
<td>4.5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>18</td>
<td>2.5</td>
<td>0.9</td>
<td>1</td>
<td>2</td>
<td>2.5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>20</td>
<td>3.8</td>
<td>0.5</td>
<td>3</td>
<td>3.75</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

### E.3 Preference Questions

<table>
<thead>
<tr>
<th>Question</th>
<th>Result:</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1</td>
<td>Offline mode is the most common response (100% of the results).</td>
</tr>
<tr>
<td>9.2</td>
<td>Online mode is the most common response (57% of the results).</td>
</tr>
<tr>
<td>9.3</td>
<td>Online mode is the most common response (71% of the results).</td>
</tr>
<tr>
<td>9.4</td>
<td>Online mode is the most common response (71% of the results).</td>
</tr>
<tr>
<td>14</td>
<td>The metadata card is the most common response (56% of the results).</td>
</tr>
</tbody>
</table>
Flopim Java Code

F.1 Abstract Input Class

```java
/**
 * This class contains all the common and abstract resources of the input classes. Each input class has to extend this one.
 */
public abstract class GlopimAbstractInput extends ServerResource {

    /**
     * Generate the response to a HTTP GET request.
     * @return a String representation of the page (HTML)
     */
    @Get
    public abstract Representation represent();

    GObject gobject;
    String gobjectName;

    protected void doInit() throws ResourceException {}

    /**
     * Generate the response to a HTTP POST request, and process the Gobject contained in the request. This GObject is sent to GlopimDispatcher where it will be transmitted to the outputs.
     * @return a String containing OK or the error which occurred
     */
    @Post
    public StringRepresentation receiveGObject(Representation r) {}
}
```
F.2 Abstract Output Class

```java
public abstract class GlopimAbstractOutput extends ServerResource
    implements Observer {

    /**
     * This class contains all the common and abstract resources of the output classes. Each output class has to extend this one.
     */
    public abstract class GlopimAbstractOutput extends ServerResource implements Observer {

    /**
     * This method processes the GObject and transmit it to the output service.
     */
    protected abstract void sendGObject(GObjectIdentified go);

    /**
     * This method gets the name of the output interface.
     */
    protected abstract String getName();

    Map<String, String> config = null;
    Map<String, String> mapping = null;

    /**
     * When the observer is notified, this method tests if the argument is a GObjectIdentified, and if it is the case calls sendGObject.
     */
    @Override
    public void update(Observable o, Object arg) {}

    /**
     * This method returns the bufferedImage (if any) serialised in the GObject.
     */
    protected BufferedImage getImageFromGObject(GObjectIdentified go) {}

    /**
     * This method tests if the current output is activated or not
     */
    private boolean isActivated() {}

    /**
     * Test if an action specifies that this output has to be used.
     */
    private boolean isSpecifiedAsReceiver(GObjectIdentified arg) {}

    /**
     * Generate the response to a HTTP GET request.
     */
    @Get
    public Representation represent() {}
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>The list.it interface</td>
</tr>
<tr>
<td>2.2</td>
<td>The web-based NotePals note browser</td>
</tr>
<tr>
<td>2.3</td>
<td>A snapshot of the MyLifeBits interface</td>
</tr>
<tr>
<td>2.4</td>
<td>A snapshot of the Haystack interface</td>
</tr>
<tr>
<td>2.5</td>
<td>The HomeViews architecture</td>
</tr>
<tr>
<td>2.6</td>
<td>A page of the notebook</td>
</tr>
<tr>
<td>2.7</td>
<td>Assigning a category to a note</td>
</tr>
<tr>
<td>3.1</td>
<td>The Glopim information workflow</td>
</tr>
<tr>
<td>4.1</td>
<td>The Glopim architecture</td>
</tr>
<tr>
<td>4.2</td>
<td>The Glopim web interface</td>
</tr>
<tr>
<td>4.3</td>
<td>The WordPress actions</td>
</tr>
<tr>
<td>5.1</td>
<td>Glopim iPaper Architecture</td>
</tr>
<tr>
<td>5.2</td>
<td>Glopim notebooks, metadata bookmark and pens</td>
</tr>
<tr>
<td>5.3</td>
<td>The feedback window</td>
</tr>
<tr>
<td>5.4</td>
<td>Send signal</td>
</tr>
<tr>
<td>5.5</td>
<td>Adding metadata using tags</td>
</tr>
<tr>
<td>5.6</td>
<td>Adding metadata using the metadata bookmark</td>
</tr>
<tr>
<td>5.7</td>
<td>Creation of a custom button</td>
</tr>
<tr>
<td>5.8</td>
<td>Adding metadata using custom buttons</td>
</tr>
<tr>
<td>6.1</td>
<td>The image box</td>
</tr>
<tr>
<td>6.2</td>
<td>The custom metadata box</td>
</tr>
<tr>
<td>6.3</td>
<td>The notes manager plugin</td>
</tr>
<tr>
<td>6.4</td>
<td>The notes manager widget</td>
</tr>
<tr>
<td>6.5</td>
<td>The faceted view plugin</td>
</tr>
<tr>
<td>7.1</td>
<td>Usefulness of Glopim in different use cases</td>
</tr>
<tr>
<td>7.2</td>
<td>Another variant of the send signal</td>
</tr>
<tr>
<td>7.3</td>
<td>Preferred method</td>
</tr>
<tr>
<td>7.4</td>
<td>Rating of methods</td>
</tr>
<tr>
<td>8.1</td>
<td>The Glopim data model</td>
</tr>
<tr>
<td>8.2</td>
<td>The Log output</td>
</tr>
<tr>
<td>8.3</td>
<td>An HTML form generated by Glopim</td>
</tr>
<tr>
<td>8.4</td>
<td>Sending information to Glopim via email</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>8.5</td>
<td>An AwareNews peripheral display</td>
</tr>
<tr>
<td>8.6</td>
<td>Handwritten content on AwareNews</td>
</tr>
<tr>
<td>8.7</td>
<td>Posting on Twitter through Glopim</td>
</tr>
<tr>
<td>8.8</td>
<td>Posting on Facebook through Glopim</td>
</tr>
<tr>
<td>8.9</td>
<td>Information transiting through two Glopim platforms</td>
</tr>
</tbody>
</table>
I would like to express all my gratitude to Adriana Ispas and Michael Nebeling for their unfailing support and plentiful advice during all the phases of this project. Moreover, I would like to thank the participants in the user study for their time and valuable feedback. I would also like to thank Prof. Moira Norrie for giving me the opportunity to work on this project within the Global Information Systems Group.
Bibliography


