SourceForge Analysis, Evaluation and Improvements

Lars Preben Arnesen
Jose Mägyä Dimiti
Jan Ingvoldstad
Andreas Nergaard
Narendra K. Sitaula

IN-DIW

2000-05-19
Contents

Preface 1

1 Introduction 3
  1.1 Research/Evaluation Goals .................................. 3
  1.2 User Group Definitions ........................................ 4
    1.2.1 Pragmatic Users .......................................... 4
    1.2.2 Political Users ........................................... 5
    1.2.3 Other Users ............................................... 5
  1.3 Findings ..................................................... 5
  1.4 Improvements ................................................. 5

2 Method 7
  2.1 Our Work, Tools, and Problems ............................... 7
  2.2 System Development Methods for Improving SourceForge .... 8
    2.2.1 The Spiral Model ......................................... 8
    2.2.2 Relating This Report to the Method ....................... 9

3 Analysis and Evaluation 11
  3.1 Strategy ........................................................ 11
  3.1.1 Intention .................................................... 11
  3.1.2 Mission Statement .......................................... 12
  3.1.3 Affiliations and Partners .................................. 12
  3.2 Communication ................................................ 14
    3.2.1 Site Introduction ........................................ 14
    3.2.2 Design Motivations ...................................... 14
    3.2.3 Page Design, Content Design and Layout ................ 15
    3.2.4 Communication with Impaired Users ....................... 16
    3.2.5 Multi-Lingual Communication ............................. 16
  3.3 Information Architecture .................................... 18
    3.3.1 Current Architecture ..................................... 18
    3.3.2 Problematic Issues and Good Points .................... 22
CONTENTS

3.3.3 Possible Enhancements ........................................... 26
3.4 Usability ........................................................................ 29
  3.4.1 Main Page ................................................................. 29
  3.4.2 Project Pages ............................................................. 29
  3.4.3 Software Map ............................................................ 30
  3.4.4 Code Snippets ............................................................ 30
  3.4.5 Search ....................................................................... 30
  3.4.6 Site Docs .................................................................... 31
  3.4.7 Discussion Forums ...................................................... 31
  3.4.8 Support Functionality ................................................ 33
  3.4.9 Personal Page ............................................................ 33
3.5 Technical Implementation .................................................. 34
  3.5.1 User Interface .............................................................. 34
  3.5.2 Servers .................................................................... 34
  3.5.3 Software ................................................................... 34
3.6 Interactivity ...................................................................... 37
  3.6.1 Transmittative Interactivity ........................................ 38
  3.6.2 Consultative Interactivity ......................................... 38
  3.6.3 Conversational and Registrative Interactivity ................. 38

4 Improvements .................................................................... 41
  4.1 Introduction ................................................................... 41
  4.2 Main Page ...................................................................... 42
    4.2.1 Design ..................................................................... 42
  4.3 Information Architecture ................................................ 47
    4.3.1 Search ..................................................................... 47
    4.3.2 Navigation, Links and Labeling ................................. 47
  4.4 User’s Personalized Work Environment .............................. 50
    4.4.1 Personal Page .......................................................... 50

5 Conclusion ......................................................................... 53
  5.1 Future Work ................................................................... 54

Bibliography .......................................................................... 55

Appendices .......................................................................... A-1

A Source Code ................................................................ A-1
  A.1 Style Sheets ................................................................. A-2
    A.1.1 Original Style Sheet ................................................ A-3
CONTENTS

A.1.2 Improved Style Sheet ................................................. A-4
A.2 PHP3 Files ................................................................. A-8
  A.2.1 Definitions ......................................................... A-9
  A.2.2 HTML Head ........................................................... A-10
  A.2.3 Session Handler ..................................................... A-12
  A.2.4 Login Status ....................................................... A-13
  A.2.5 Navigation Class .................................................... A-14
  A.2.6 Global Navigation Handler 1 ..................................... A-15
  A.2.7 Global Navigation Handler 2 ..................................... A-18
  A.2.8 Local Navigation Handler ......................................... A-20
  A.2.9 Page Header .......................................................... A-25
  A.2.10 Content Base ....................................................... A-32
  A.2.11 Page Footer ........................................................ A-34
  A.2.12 Standard Page Base .............................................. A-36
  A.2.13 Main Page .......................................................... A-37

B SourceForge Hardware .............................................. B-1
### List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Top navigation bar</td>
<td>18</td>
</tr>
<tr>
<td>3.2</td>
<td>Second top navigation bar</td>
<td>19</td>
</tr>
<tr>
<td>3.3</td>
<td>Left navigation bar</td>
<td>20</td>
</tr>
<tr>
<td>3.4</td>
<td>IA graph — main page</td>
<td>23</td>
</tr>
<tr>
<td>3.5</td>
<td>IA graph — discussion forums</td>
<td>24</td>
</tr>
<tr>
<td>3.6</td>
<td>Discussion Forum: Help (main page, 1/2)</td>
<td>24</td>
</tr>
<tr>
<td>3.7</td>
<td>Discussion Forum: Help (main page, 2/2)</td>
<td>25</td>
</tr>
<tr>
<td>3.8</td>
<td>Discussion Forum: After choosing the first thread in 3.6 (not logged in)</td>
<td>25</td>
</tr>
<tr>
<td>3.9</td>
<td>Discussion Forum: Same as 3.8, but logged in</td>
<td>26</td>
</tr>
<tr>
<td>3.10</td>
<td>Discussion Forum: After choosing the thread again</td>
<td>27</td>
</tr>
<tr>
<td>3.11</td>
<td>Discussion Forum: After choosing “Save Place”</td>
<td>32</td>
</tr>
<tr>
<td>3.12</td>
<td>Discussion Forum: After choosing “Monitor Forum”</td>
<td>32</td>
</tr>
<tr>
<td>3.13</td>
<td>Discussion Forum: After choosing “Monitor Forum” again</td>
<td>32</td>
</tr>
<tr>
<td>4.1</td>
<td>New Main Page Design, Anonymous User</td>
<td>42</td>
</tr>
<tr>
<td>4.2</td>
<td>New Main Page Design, Anonymous User (2)</td>
<td>43</td>
</tr>
<tr>
<td>4.3</td>
<td>Simpler Main Page Design, Anonymous User (3)</td>
<td>43</td>
</tr>
<tr>
<td>4.4</td>
<td>Simpler Main Page Design, Anonymous User (4)</td>
<td>44</td>
</tr>
<tr>
<td>4.5</td>
<td>New Main Page Design, Logged In User</td>
<td>44</td>
</tr>
<tr>
<td>4.6</td>
<td>New Main Page Design, Logged In User (2)</td>
<td>45</td>
</tr>
<tr>
<td>4.7</td>
<td>Simpler Main Page Design, Logged In User (3)</td>
<td>45</td>
</tr>
<tr>
<td>4.8</td>
<td>Improved Information Architecture</td>
<td>48</td>
</tr>
<tr>
<td>4.9</td>
<td>Simpler Main Page Design, New Search Option (Site Search)</td>
<td>49</td>
</tr>
<tr>
<td>4.10</td>
<td>User Home Page</td>
<td>50</td>
</tr>
<tr>
<td>4.11</td>
<td>Monitoring Settings (1)</td>
<td>51</td>
</tr>
<tr>
<td>4.12</td>
<td>Monitoring Settings (2)</td>
<td>52</td>
</tr>
</tbody>
</table>
Preface

We are a group of graduate students at the Department of Informatics (University of Oslo, Norway), studying Communication Systems and System Development. This document is a report of our work in the three-credit course IN-DIW – Design of Interactive Web-services.

Chapter overview

Chapter 1 – Introduction introduces SourceForge, the project, and our viewpoints.

Chapter 2 – Method explains the methods we set out to use, and what we actually ended up doing.

Chapter 3 – Analysis and Evaluation describes the analysis and evaluation of SourceForge.

Chapter 4 – Improvements describes the improvements we suggest for SourceForge, and the prototype (mock-up) we made.

Chapter 5 – Conclusion is where we present our conclusions.
Chapter 1

Introduction

SourceForge is an Open Source [Open Source Initiative, 2000], [Raymond, 1998] software hotel and distribution channel, with extensive support functionality (e.g., storage, version control, bug tracking, support systems, discussion forums). SourceForge was started in late 1999.

Our project is about analysing, evaluating and improving SourceForge as an interactive web site for open source developers.

1.1 Research/Evaluation Goals

The main goal was to find out what SourceForge has done well and what they have done badly, and make a suggestion for an improved community site.

In the analysis process, we looked at the project from five different angles (why is explained in chapter 2):

1. Strategy (section 3.1)
2. Communication (section 3.2)
3. Information Architecture (section 3.3)
4. Usability (section 3.4)
5. Technical Implementation (section 3.5)

In addition, we looked at the site’s interactivity characteristics in the Interactivity section (3.6).

Improvements on our part were done by changing communication, information architecture, usability and interactivity aspects of the site and implementing them in a prototype mock-up (described in chapter 4).
1.2. USER GROUP DEFINITIONS

Both for analysis and improvement, we had to define SourceForge's user groups.

1.2 User Group Definitions

There are several user groups, with different motivations for their use of the web site. We have defined the following user groups, inspired by [Nergaard, 1999], [Hannemyr, 1999] and [Raymond, 1998].

1.2.1 Pragmatic Users

The Core Developer

The core developer is a member of the central group of programmers on a project, with a need for full access to the entire project. Other activities include writing code snippets (see section 3.4.4) for the code snippet library.

The developer/advanced user

The advanced user is a user who follow the development of a project closely, and typically want to test the latest non-release version of the program, as well as modify the code for their own use. They may also be interested in contributing suggestions for new features and bug fixes.

The "regular" end user

The end user is an existing user of the program who is interested in new releases and support, without interest in taking part in discussions or the debugging process.

The browser

The browser is a user who looks for a program that may fit her needs, and browses different projects to find the best.

Note: In the two last categories we can separate end-users and browsers with and without deep technical knowledge. For some projects, this should be considered a major issue, but for other projects, such as programming tools, only one of these groups will be an interesting user group.
1.2.2 Political Users

Many Open Source projects have a political motivation as well. Therefore we might want to look at users interested in advocacy (open source vs closed source, [Raymond, 1998]) and political/philosophical discussions, and users who just want the best software.

1.2.3 Other Users

There may be other users of SourceForge than those mentioned above, but they are probably not in the intended target audience [Raymond, 1998].

1.3 Findings

During analysis and evaluation, we found several problems with the current implementation of SourceForge, and the most prominent were in terms of information architecture and usability. The site did not seem to be very well planned, and web pages were often more difficult to navigate than necessary.

1.4 Improvements

The most striking improvements done were grouping global navigational elements together, separating local and global navigation, changing navigation labels and slight architectural changes by moving documents around in the site hierarchy.
1.4. IMPROVEMENTS
Chapter 2

Method

2.1 Our Work, Tools, and Problems

When we started, we realized that if our work was to be efficient, we had to divide it. We decided to find complimentary points of view, and since we did not want a technical focus on our work, we separated the technical issues from the rest. We wanted to look into how SourceForge was a part of a larger context, and chose to look into the strategic aspects of the site. We also wanted our major focus to be on usability, and therefore we let this be a point of view, backed up by design issues and information architecture.

We have worked quite loosely organized, with quite a lot of individual freedom. Each group member were given primary responsibility for one point of view, and we coordinated our efforts by having meetings once or twice per week. If additional communications was needed, we used e-mail or telephone. Important issues from the meetings were also sent out by e-mail. CVS\(^1\), a version control system, was used to store all our work. The report was written in \LaTeX\(^2\). The prototype was coded in HTML\(^3\) and CSS\(^4\), using PHP\(^5\) as a back-end. Generation of documents and the web site was done through batched jobs locally and remotely.

We faced a few problems during our work. As most of us have jobs needing attendance, it was sometimes difficult to coordinate meetings where everyone was present. The deadlines we had set also crashed with deadlines at work or other courses. Two of the group members had family issues consuming a lot of time at critical points in the process, and as we had quite a lot of

\(^1\)Concurrent Versions System, [SourceGear, 2000]
\(^2\)Lamport’s TeX, [The \LaTeX\ Project, 1999]
\(^3\)Hyper Text Markup Language, [World Wide Web Consortium, 1999]
\(^4\)Cascading Style Sheets, [Lie and Bos, 1996]
\(^5\)Professional Home Pages, [PHP, 2000]
individual freedom, our project sometimes suffered from low priority.

Our biggest problem, however, was the lack of response from the SourceForge team. They were initially enthusiastic about our project, and promised us help and guidance. However, when we contacted them later, they did not respond. As we had assumed cooperation from them when planning our work, our project suffered severely, and we had to make assumptions about some issues.

There have been differences of opinion about design, architecture, interactivity and implementation issues, but none that we didn’t resolve.

2.2 System Development Methods for Improving SourceForge

As SourceForge is part of the open source movement, it is only to be expected that it is developed as an open source project, with quick iterations and heavy reliance on user feedback [Raymond, 1998]. However, since SourceForge in itself is critical for the open source movement (after all a lot of other projects rely on it), a more formal method should be implemented.

We suggest the spiral Model [Boehm, 1988] for the SourceForge development, but less formalized to fit the mindset of the hacker community. Rapid adaption to new needs may be crucial to keep the users and projects.

2.2.1 The Spiral Model

The spiral model has, put simply, four phases which repeat, thereby forming a spiral, which at a certain point in each turn leads to a new version of the product. The four phases are:

- Plan next phases
- Determine objectives, alternatives, constraints
- Evaluate alternatives, identify and resolve risks
- Develop and verify next-level product

These, however, are just principles to build a more specific method on. We suggest to do plan the steps as follows.
CHAPTER 2. METHOD

Plan next phases  This step is perhaps the one where open source development has its biggest advantages. The direct contact with the users and the collective way of developing new features makes open source projects more dynamic when it comes to implementing new features on demand. [Raymond, 1998]

SourceForge supports this by their “Request New Feature”-links on the main page. Here, users can submit wishes or needs, and with discussion forums for the site for evolving ideas on how features should work.

Determine objectives, alternatives, constraints  Determining objectives and alternatives can be done as a community effort, by discussing features in the public forums. However, at a certain point this discussion should be restricted to those who have intimate knowledge of the system. This group are the ones most competent when it comes to identifying constraints.

Evaluate alternatives, identify and resolve risks  The project team should then choose which alternative to implement, and by applying risk analysis make the development of the new features more feasible. In this step, prototypes should be used to test new features.

Develop and verify next-level product  In this step, the development itself takes place. Every feature should be properly tested before release.

[Norin and Stöckel, 1998] claims that the spiral model is not suited as a model for open source software development, because the risk analysis is often skipped. We can agree with this in most cases, but in critical systems, as SourceForge has become to the open source community, risk analysis should be on the development team’s agenda.

2.2.2 Relating This Report to the Method

What we have done in this project is covered by the first three steps. Chapter 3 covers the planning step, where we identify problems with the site. While developing the prototype (chapter 4), we went through the second steps by determining objectives and considering alternatives. In the prototype, we also chose preferred alternatives, and implemented dummies. Considering technical constraints and identifying technical risks was not possible, so we chose to create improvements as if no significant technical constraint or risks existed.
2.2. SYSTEM DEVELOPMENT METHODS FOR IMPROVING SOURCEFORGE
Chapter 3

Analysis and Evaluation

This chapter presents what SourceForge was like in March, 2000, and our analysis and evaluation of the state of the site at that time. The site has had some minor changes since then, which is only to be expected for a young and dynamic site.

3.1 Strategy

3.1.1 Intention

In this section, we analyse the strategy behind SourceForge; what are their goals, their policies, and why. For instance, why did they choose their current geographical location, which effectively hinders cryptographic projects’ participation? This question is left unanswered, we can only guess. A plausible reason is that VA Linux, SourceForge’s main sponsor and creator, is California based, and that they didn’t want to sponsor computers in another country.
3.1. STRATEGY

3.1.2 Mission Statement

SourceForge’s mission is to enrich the open source community by providing a centralized place for open source Developers to control and manage open source software development. [SourceForge, 2000a]

We intend to analyze this statement according to

- How the site is today
- The open source community ideals
- Possibilities and restrictions posed by the mission statement

SourceForge already hosts a number of interesting projects, and therefore contributes to distribution of open source, and therefore also to the community as a whole. It follows Eric S. Raymond’s advices for running an open source project by implementing vital features such as communication lines between different kinds of users [Raymond, 1998]. By enabling users to monitor projects, they also give users an incentive to follow certain projects closely, which is important in enabling the different projects to evolve into separate communities.

After some time of cleaning up the code, SourceForge’s own source code was released to the public under the GNU GPL 1 January 14 2000, making a clear statement about the commitment to the open source community.

The mission statement is very open, in the sense that it poses very few restrictions on their work. It gives the site managers the freedom to implement whatever they see the need for at the present time.

3.1.3 Affiliations and Partners

Affiliations

SourceForge is affiliated with and developed by VA Linux, a software company specializing in e-commerce solutions based on the Linux platform. VA Linux sponsors SourceForge with both manpower and hardware.

1GNU General Public License, [Free Software Foundation, 1991]
Partners

SourceForge has the following partners:

**Cosource.com**

“a collaborative, reverse-auction web site enabling international consumers and developers of open source products to work together to fund development of innovative software solutions.”

[SourceForge, 2000b]

**linux.com**

Linux.com’s mission is to enrich the Linux community by providing a centralized place for individuals of all experience levels to learn (and teach) the power and virtues of the Linux Operating System.

[Linux.com, 2000]
3.2 Communication

In this section we focus mainly on aspects related to the communication between SourceForge and its diverse groups of users or customers. (For a more detailed definition of the different groups of users, please refer section 1.2.) Obviously we consider these aspects with the user's needs in mind and SourceForge being the focal point of reference. This certainly leads us to the issue of usability – being inherent or absent in this very communication.

Communication in this context basically refers to the means by which the customer and SourceForge exchange information. We look at the way in which SourceForge imparts information to and captures useful information from its users.

Page design, content design, page layout, consideration for users with different abilities, multi-lingual capabilities of communication, etc are the aspects of communication with SourceForge’s users we’ll be looking at.

The user most probably desires appealing communication, effective communication and user-friendly communication. In order to muster mutual trust, the user also considers convenient and timely communication to be of utmost relevance and importance [Nielsen, 1999].

3.2.1 Site Introduction

A special variant of user, the ordinary browser or surfer, perhaps accidental, who has had little or no exposure to the open source software movement or SourceForge probably immediately raises the question as to what SourceForge is all about. Perhaps what SourceForge communicates on its main page may bewilder that particular user as she ponders over the obfuscation inherent in the overall site and the links, especially on the right-hand column, in addition to the Latest News column. To another variant of user, however, say a member of the development team or a seasoned open source advocate or participant, the site communicates perfectly well. Does this lead us to question the timely communication of information by SourceForge to the diverse user space? On re-evaluation however, we may consider that since the occasional "accidental" browser user group may be limited perhaps it warrants less concern in terms of timely communication in that regard.

3.2.2 Design Motivations

When we consider communication between SourceForge and its users, perhaps it will help to consider upfront the apparent underlying political motivations as the driving force behind the particular choice of design and layout.
CHAPTER 3. ANALYSIS AND EVALUATION

The reader should be aware of the mindset behind the different groups of users, developers, and not least the administrators of the website. This is clearly reflected in the design of the site. The use of IRC\(^2\)-like nicknames\(^3\) that bewildered the occasional surfer in the previous section further reflects and enforces this observation. This is one of the very reasons why some of SourceForge’s users are classified under political users (see section 1.2.2).

3.2.3 Page Design, Content Design and Layout

Cross-Platform/Environment Design

Objectively, the user who prefers or must use a text-based hypertext browser such as Lynx may suffer inconvenient communication due to firstly, the inevitable inherent nature of text-based hypertext browsing generally over a wide area network such as the Internet. Secondly, and more relevant to our scenario, the lack of cross-platform page design to accommodate text-based browsing. A case in point is the Code Snippets page. Snippet title and author tend to be cluttered into a sentence such that an oblivious user may take sometime to decode the information. Furthermore, as mentioned earlier, the excessive usage of IRC-like nicknames to denote contributors and code authors, though very much desirable in many discussion forums on the Internet may not necessarily appeal to the pool of diverse users interested in SourceForge. But, as also mentioned, this still reflects the open source mindset.

Color and Graphics

One way to facilitate convenient and seamless communication between a site and its users is that the choice of good color and graphics is of crucial importance [Nielsen, 1999], [Dix et al., 1993]. SourceForge tends to excel in this area, with the choice of professionally appealing color consistent throughout a page and its sub-pages, the ample usage of white space, the less cluttering usage of graphics. Portable Network Graphics (PNG)\(^4\) is the graphic format SourceForge has chosen, rather than formerly more common encoding

\(^2\)Internet Relay Chat

\(^3\)IRC nicknames are on the form “DareDevil”, or some other simple nickname the user chooses to represent her while online

\(^4\)PNG is a portable storage format for raster graphics designed to work well with online applications such as the Web. It is designed as a replacement for the picture formats GIF and TIFF (see also [Thomas Boutell, 1997])
formats such as the proprietary GIF\textsuperscript{5} or JPEG. As mentioned, this reflects the underlying “liberation” as well as the “ease of use” mindset (open source) existent in the large portion of the SourceForge user space.

### 3.2.4 Communication with Impaired Users

SourceForge does not rely on the use of sound to give feedback on users’ actions. Feedback comes solely from the use of text and inline graphics, yet without being dependant on the graphics; all icons and images have decent ALT attributes set, and except for the label weaknesses mentioned in the information architecture section (3.3), there are few obvious hindrances and nuisances for visually impaired users. Colors almost seem like they have been chosen with color blind people in mind.

All significant colors are in shades of blue and gray (very few color blind people have problems with blue, red-green color blindness is most common, [Dix et al., 1993], [Nielsen, 1999]). The only major downside to color choice is that in the second top navigation bar, there is little contrast between link colors and background color. Another possible problem is that the page layout is made with tables, which might get rendered poorly by text-only browsers (according to our studies, Lynx doesn’t do tables very well), or cause problems for braille reading strips or voice synthesis software.

### 3.2.5 Multi-Lingual Communication

Besides a few translated “HowTo”- and “Getting started”-documents SourceForge bases its entire site on English. This clearly is a drawback for those users who don’t understand English, but the majority of the open source projects has standardized on English, which makes it a natural choice for an open source community site. A translation of the SourceForge site would make it easier for non-English users to create a project and use the services that SourceForge provides, but the browser would still have problems since the majority of the projects hosted by SourceForge use only English. It’s not likely that every project would translate all the documentation/information to several languages, but SourceForge could implement translation tools like BabelFish\textsuperscript{6} to dynamically translate all information before it’s presented to the user. Translation tools like BabelFish are quite young, don’t provide

---

\textsuperscript{5}GIF uses the LZW compression algorithm, which is owned in full by Unisys. Unisys has followed through on their patent [Unisys, 2000], and made claims for millions of dollars against web sites around the world ["Justin", 1999], ["Roblimo", 1999], ["Jamie", 2000].

\textsuperscript{6}BabelFish is a translation service delivered by AltaVista that translates text snippets or even whole web pages between the most common languages. [Systran, 2000]
100% correct translations, and they still don’t provide support for more than a few major languages\textsuperscript{7}.

\textsuperscript{7}In the case of Babelfish, English, French, German, Italian, Portuguese and Spanish
3.3 Information Architecture

In this section, we’ll present, analyse and evaluate SourceForge’s architecture, with focus on navigation and labels.

3.3.1 Current Architecture

Navigation and Labels

SourceForge has two slightly different global navigation sets [Rosenfeld and Morville, 1998]: one for registered users who are logged in, and one for all the others. This affects the apparent architecture as seen by the users. Both sets have enough in common that we’ll treat them as the same for now.

The navigational elements treated later in this section are the same for all SourceForge pages that the authors have browsed through. However, when browsing code snippets, packages or other software, additional navigation options appear in the content area of each page.

Global Navigation

The navigation bars are implemented by using relatively plain HTML 4 [World Wide Web Consortium, 1999], although not entirely standards compliant\(^8\) combined with style sheet effects, which means that there are no frames attempting to control the navigation layout or keep things “in place”. This means that all the navigational options are available at any place, even if the user jumps in via an external link to any page.

As an anonymous user, we find three navigation bars, two at the top (figures 3.1 and 3.2) and one to the left (figure 3.3). Most navigational elements are textual, only two are iconic labels [Nielsen, 1999], [Rosenfeld and Morville, 1998].

\(^8\)Some attributes, such as those for \(<\text{BODY}>\), aren’t legal according to the W3C\(^9\) spec
The topmost bar (see figure 3.1) contains global navigation elements with the easily recognizable labels:

- Home
- About
- Partners
- Contact Us
- Login (changes to Logout when the user is logged in)

The “Home” label points back to the main page, and there it’s self-linked, which is redundant because the same link appears with the SourceForge logo in the next navigation bar. Similar self-linking appears when the user visits either of the pages local to sourceforge.net available in this navigation bar. Self-linking is confusing to the users, because it creates the illusion that the link points to somewhere else than the current page, while following the link will lead the user nowhere [Nielsen, 1999].

In addition, there is a label “linux.com Partner”, which is used for linking to linux.com’s web site. (Curiously enough, this link is not listed as a partner on the Partners page.)

The second bar (see figure 3.2) is a combination of a navigation bar and a status field. It is here we find the only two iconic link labels used in global navigation, one for SourceForge itself (links back to the home page, self-linked there, but missing an ALT attribute) and one for VA Linux Systems (includes descriptive ALT attribute), the site’s main sponsor.
3.3. INFORMATION ARCHITECTURE

Figure 3.3: Left navigation bar

The textual navigation has the labels:

- [Login] (disappears when logged in)
- [New User] (disappears when logged in)
- [Software Map] (self-linked)
- [New Releases] (self-linked)
- [Site Docs] (self-linked)
- [Top Projects] (self-linked)

[Login] is redundant with the same link found in the top navigation bar.

The third bar (see figure 3.3) is placed on the left side, and holds more project related navigation, divided into three categories: SourceForge, Not Logged In/Logged In: username (shows status) and Search.
Starting with the SourceForge category, the labels are:

- Homepage (self-linked, redundant)
- Code Snippet Library (self-linked)
- Software Map (self-linked, redundant)
- New Releases (self-linked, redundant),
- Site Documentation (self-linked, redundant),
- Top Projects (self-linked, redundant),
- Other Site Mirrors (self-linked, surprisingly not redundant)

It is rather difficult to see where a link begins and ends; the user has to guess which words belong together.

In the login category, we have:

- Login via SSL
- New User via SSL

We see these in both the other navigation bars, although here it informs us that we’ll be using SSL.\(^\text{10}\)

For searching, there is a small search form with a selection menu. The form contains the following items:

- Option: Software/Group
- Option: People
- A checkbox (“Require All Words”)
- A free text search field
- A search button

There is no option for searching the site itself, although the forums (feature request, help and discussion) allow for search of “This Forum”. It would be beneficial if it was possible to search the part of the site one was currently in, for instance “Site Documentation” [Nielsen, 1999].

\(^{10}\text{Secure Sockets Layer, see [whatis.com, 1999]}\)
Architecture Graphs

As one can see from figure 3.4, SourceForge has a pretty complex site layout, if we look at it from a new or anonymous user’s point of view (the view is slightly more complex if the user is logged in). There are lots of links to other pages, several are referred multiple times.

In addition to the other apparently important pages, there are several dynamically updated embedded links [Rosenfeld and Morville, 1998] in the content part of the main page, such as news bulletins, usage statistics, popular, new and ranked projects. Figure 3.5 shows the link relationships in an IA graph [Rosenfeld and Morville, 1998] when dealing with discussions in SourceForge.

If a user wants to read articles in a discussion forum, she will have to locate the relevant forum from either the main page (which is what we illustrate in figure 3.5, visualized with the screen shot in figure 3.6 and 3.7), or from a particular project’s page.

Then she must choose the thread/subject of interest, not only once, but twice before she is allowed to read the message, yet it’s possible to write a response before reading any messages (figures 3.8 and 3.9).

3.3.2 Problematic Issues and Good Points

Already at the main page, we see severe navigational problems. There are too many navigation links and several redundant ones with different labels for the same links. There is also a first case of self-linking; the main page points to itself in three different places. These things result in a generally bad site structure [Nielsen, 1999], [Rosenfeld and Morville, 1998].

The internal structure of the page elements appears to be in order, except that “Help and discussion” is hidden far down on the page instead of being linked to in navigation bars.\footnote{Some extensions to the model in [Rosenfeld and Morville, 1998] have been made.}

Not all labels are as intuitive as they should [Nielsen, 1999], [Rosenfeld and Morville, 1998]. What does “Software Map” mean? If we’re just curious, we would find out that this is a categorical hierarchy of the software projects – internally alphabetical – as well as an alphabetical list of some assorted software.\footnote{This changed after we published our analysis document on the web and made it known to the SF team, but it may be part of an independent reorganization of the main page.}

\footnote{The software map was also changed to a new version, but the points mentioned above are still valid.}
CHAPTER 3. ANALYSIS AND EVALUATION

Figure 3.4: IA graph — main page
3.3. INFORMATION ARCHITECTURE

![SourceForge Information Architecture Diagram]

**Figure 3.5: IA graph — discussion forums**

**SourceForge — Message Forums**

**Discussion Forums: Help**

<table>
<thead>
<tr>
<th>Thread/Subject</th>
<th>Author</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>#20 Part 500 Error</td>
<td>arain</td>
<td>03/22/00 05:53</td>
</tr>
<tr>
<td>#21 SCP for Macintosh</td>
<td>numb</td>
<td>03/20/00 08:41</td>
</tr>
<tr>
<td>#22 Problem with mail archives</td>
<td>jump</td>
<td>03/20/00 06:27</td>
</tr>
<tr>
<td>#23 ssh wont let me in</td>
<td>Kal-Le</td>
<td>03/19/00 22:54</td>
</tr>
<tr>
<td>#24 No</td>
<td>edream</td>
<td>03/19/00 10:03</td>
</tr>
<tr>
<td>New admin account does not grant access</td>
<td>steponte</td>
<td>03/19/00 19:06</td>
</tr>
<tr>
<td>#26 cookies on sourceforge</td>
<td>philhoward</td>
<td>03/19/00 13:14</td>
</tr>
<tr>
<td>Download Count</td>
<td>twalker</td>
<td>03/19/00 00:34</td>
</tr>
<tr>
<td>cache.php probe with local SF install</td>
<td>jntu</td>
<td>03/17/00 16:38</td>
</tr>
</tbody>
</table>

**Figure 3.6: Discussion Forum: Help (main page, 1/2)**
CHAPTER 3. ANALYSIS AND EVALUATION

SourceForge – Message Forums

Discussion Forums: Help

Expanded View | Monitor Forum | Save Place | Post | Admin

<table>
<thead>
<tr>
<th>Thread/Subject</th>
<th>Author</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perl 500 Error</td>
<td>aran</td>
<td>03/22/00 08:33</td>
</tr>
<tr>
<td>SCP for Macintosh?</td>
<td>numb</td>
<td>03/21/00 08:41</td>
</tr>
<tr>
<td>Problem with mail archives?</td>
<td>jorit</td>
<td>03/20/00 06:27</td>
</tr>
<tr>
<td>SSH won’t let me in</td>
<td>Kal-Le</td>
<td>03/19/00 22:54</td>
</tr>
<tr>
<td>ftp</td>
<td>edream</td>
<td>03/19/00 10:39</td>
</tr>
<tr>
<td>New admin account does not grant access</td>
<td>stepone</td>
<td>03/19/00 19:08</td>
</tr>
<tr>
<td>cookies on sourceforge</td>
<td>philhoward</td>
<td>03/19/00 13:14</td>
</tr>
<tr>
<td>Download Count...</td>
<td>twoker</td>
<td>03/19/00 00:34</td>
</tr>
<tr>
<td>cache.php probe with local SF install</td>
<td>jantu</td>
<td>03/17/00 16:38</td>
</tr>
</tbody>
</table>

Figure 3.7: Discussion Forum: Help (main page, 2/2)

SourceForge – Message Forums

Discussion Forums: Help

Expanded View | Monitor Forum | Save Place | Post | Admin

<table>
<thead>
<tr>
<th>Thread/Subject</th>
<th>Author</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perl 500 Error</td>
<td>aran</td>
<td>03/22/00 08:33</td>
</tr>
</tbody>
</table>

Post to this thread

You could post if you were logged in

Figure 3.8: Discussion Forum: After choosing the first thread in 3.6 (not logged in)
Even though the navigational structure leaves much to be desired, there are several redeeming points. Most of the important links are placed in positions where they are easy to find, and new users shouldn’t have any problems finding a way to register themselves if they so desire.

The search function provides an easy interface for finding a particular project if the user doesn’t want to spend time guessing at which label is the right one, but that is only a patch for a bad site architecture [Nielsen, 1999].

There is a downside, however, as the search interface is difficult to understand. Search by software/group lets you search by type of software, for instance “mail encryption” or by the name of a “group” (in this context meaning “software project name”, but how can we guess that?). Search by person lets you search by who’s participating in a project. All of these result in a list of matching projects.

### 3.3.3 Possible Enhancements

The site’s navigational structure would improve a lot if there were far fewer redundant links, and if relevant links such as those for documentation and bug reporting were in a more prominent position on the page, for instance.
Figure 3.10: Discussion Forum: After choosing the thread again
in a navigation bar [Nielsen, 1999].

The “Software Map” label could benefit from a rephrasing, for instance into “Software Categories” or “Software By Category”, which would make it easier to find this certainly useful feature for the users who are looking for particular kinds of software.

Message forums would benefit a lot from better labels, fewer navigation steps and maybe more space than the current global navigation bars allow.
3.4 Usability

In general, the site seems effective, producing the desired result: being a popular site among open source developers. Server response is good, and in spite of tables being used for layout on each page, they load reasonably fast. There are few (if any) dead links, and the main navigational elements are consistent all over the site, no matter where the user enters.

However, the regular navigational elements take up vast amounts of screen real estate (leaving only 45% for content at 800x600 and 55% at 1024x768 with Netscape Navigator 4.72). [Nielsen, 1999] disapproves of navigation taking up much of the screen real estate, but with SourceForge’s special audience (not dissimilar to the open source activists found at Slashdot [Slashdot, 2000]) probably means that the users are better equipped to handle this distribution of information [Nielsen, 2000], [Nielsen, 1999].

Internally, the navigational elements are slightly inconsistent in use of labels and link colors, and a few of the labels are difficult to separate from each other. When navigating, there is virtually no indication of where you are within the site’s hierarchy; the user will have to remember. The color contrasts are mostly good, but in some places, contrast is close to being too low [Dix et al., 1993].

3.4.1 Main Page

The main page is cluttered with information, making it hard to quickly locate information if the user isn’t familiar with the site, and so it isn’t very well suited for new users or users outside of the intended target audience [Nielsen, 1999].

In addition, there are several links (such as FAQs, help documentation and access to discussion forums) in the content part of the page that should rather be in navigational elements.

It looks like the main page is tailored for regular visitors with a task-oriented view. The headlines “What’s new?” and “What’s hip?” are relevant for a task oriented, regular user whose primary concern is to get information fast. For a new user, however, these sections only contribute to cluttering the screen.

3.4.2 Project Pages

SourceForge uses the same structure on each central web page. This makes it easy to find navigational elements, and contributes to making the site browsable [Nielsen, 1999].
Icons for diverse “actions” in the top right corner of each project subpage are small and difficult to understand at first, but standardized.

### 3.4.3 Software Map

The Software map is confusing for new users. Implementing both a “list all” and a hierarchy may be a good idea, to provide different entry points to the projects [Nielsen, 1999], [Rosenfeld and Morville, 1998].

Only the hierarchy seems to be implemented, there is an alphabetical list of software on each page in the software map, but it only seems to contain some software packages, without any explanation to which category they belong to, or if they don’t belong to any yet.

For each software package, there is a helpful feature: Miniaturized operating system (or environment) logos.

When choosing a hierarchy, an important point is whether it is scalable or not. The hierarchy in the software map looks quite balanced (which is good), but it is always difficult to tell in which direction the greatest mass of new projects will be added.

### 3.4.4 Code Snippets

Code snippets are smaller pieces of code that the developers don’t think deserve a project of their own. These are usually small utilities.

Submitting and changing code snippets is a bit of a hassle for the developer (or advanced user). The only way to submit or change the code is to cut-and-paste the new code into a text area field (in an HTML FORM [World Wide Web Consortium, 1999]) on the web side. It isn’t possible to submit a simple patch, nor edit the code interactively. This makes the whole process of dealing with code snippets, including simple bug fixes, tedious at best. A better solution would be to provide some sort of upload functionality (for instance, INPUT TYPE="file" [World Wide Web Consortium, 1999]) for code snippets that doesn’t rely on the user’s ability to cut-and-paste correctly, and to show the code in the text area field when someone has decided to put in a new version or a patch.

### 3.4.5 Search

Divided into two types of searching (People and Software/Group), the search feature provides another way to access projects (but only projects). It is quick, but lacks the ability to specify meta data (operating system, etc), and
it isn’t specified whether logical operators are possible or not. The sorting of results looks like a “most popular first”, but that’s not stated explicitly.

A site-wide free text search is often considered an important feature [Rosenfeld and Morville, 1998], [Nielsen, 1999] when the site contains a large amount of information. SourceForge lacks this, and this makes it impossible to use the search engine to find answers to e.g. support questions directly from the search form. When the information architecture is weak, search becomes more important, and we feel that this should be a major issue for the SourceForge developers.

3.4.6 Site Docs

This label is slightly misleading. There is too little information in the site documentation pages, although other information about SourceForge is available from other links on the main page. Under Site Docs, there is also a link to the SourceForge documentation project [SourceForge, 2000c]. Effectively, the Site Docs page is a subset of the documentation project.

3.4.7 Discussion Forums

After looking at the main discussion forums (SourceForge also calls them “message forums”), we decided that the help discussion forum is representative for SourceForge’s own discussion forums and local navigation in general.

Near the top right of the body part of the page, we find a lot of small icons, obviously with some special meaning. Jakob Nielsen points out in [Nielsen, 1999] that icon metaphors are seldom successful, yet these seem to serve well as labels for the links they represent, though perhaps not at first glance. Well chosen ALT attributes clears up any confusion, at least in browsers that display them, and we find that they lead us to various forums and seemingly relevant documents.

The other labels aren’t very descriptive. What does “Monitor Forum” and “Save Place” (see figure 3.11) mean? We can guess that “Expanded View” will give us more details in some way, at least. What does the check mark next to “Monitor Forum” mean? Following the link (figure 3.12), we get a message telling us that the forum is monitored, and we’ll be “emailed followups to this entire forum”. So “Monitor Forum” basically means “Get followups to this entire forum”, which isn’t very intuitive. Selecting Monitor Forum again tells us that we “will not receive any more emails from this forum” (figure 3.13). The check mark remains, though. This is obviously a bug in something that was supposed to show the state of the monitoring.
3.4. **USABILITY**

**Figure 3.11:** Discussion Forum: After choosing “Save Place”

**Figure 3.12:** Discussion Forum: After choosing “Monitor Forum”

**Figure 3.13:** Discussion Forum: After choosing “Monitor Forum” again
3.4.8 Support Functionality

New user creation is a process in several steps, which includes sending of confirmation e-mail to the e-mail address the user specifies, followed by web-based confirmation of registration by the user. This may seem complex, but the new users see good instructions and explanations, making the entire creation process efficient while creating a sense of security.

3.4.9 Personal Page

SourceForge provides a Personal Page for registered users, which is like a personalized portal into the parts of SourceForge the user is active. The Personal Page contains the following sections:

- **My Bugs** All bug reports submitted by or assigned to the user.
- **Monitored Forums** The forums the user has chosen to monitor.
- **Monitored File Modules** Shows the files the user has chosen to monitor.
- **My Tasks** Tasks assigned to the user.
- **Quick Survey** A few questions about the user’s technical skills and interest in open source projects. After answering the questions, the only contents of this section are the words “You have taken your developer survey”, which is of no practical use to the personalized page.
- **My Projects** This section lists the projects the user is a member of.

The problem with this page is that there is little information on how to actually customize the page to the user’s needs. The actual customization is done in the different sections of SourceForge. For instance, the user can choose to monitor a forum, which places that forum in the list of monitored forums in the Personal Page.
3.5 Technical Implementation

To analyze an implementation, it’s necessary to know the design model. The code itself doesn’t give the same information.

“Show me your [code] and conceal your [data structures], and I shall continue to be mystified. Show me your [data structures], and I won’t usually need your [code]; it’ll be obvious.”

Actually, he said “flowcharts” and “tables”. But allowing for thirty years of terminological/cultural shift, it’s almost the same point. [Brooks, 1975, in [Raymond, 1998]]

We were unable to get a copy of SourceForge’s white paper, which made it difficult to analyze how the site was constructed. Since the exact technical implementation is not in the scope of this report, we chose not to analyze the site’s source code.

3.5.1 User Interface

The SourceForge user interface is web based. It provides interfaces to the databases, mailing lists, CVS repositories and file download. The entire site, with the exception of a few documents, is generated with PHP.14

They also offer CVSWeb, a script that gives the users a read-only web interface to the CVS repository. This is only useful for browsing the CVS-tree without downloading the whole project.

3.5.2 Servers

The servers, sponsored by VA Linux Systems, are running the Red Hat Linux operating system. SourceForge uses a virtual server from the Linux Virtual Server Project [Linux Virtual Server Project, 2000] for making the cluster of servers look like one single server to the user. See appendix B for the list of hardware used in SourceForge.

3.5.3 Software

SourceForge have developed their site by using widely recognized open source software and then developed their user interfaces and services on top of that.

14 PHP is a server-side, cross-platform, HTML embedded scripting language.
CHAPTER 3. ANALYSIS AND EVALUATION

Database

SourceForge has based it’s site on the MySQL [MySQL, 2000] database. MySQL is a SQL\textsuperscript{15} database developed by the Swedish company T.c.X. MySQL is open source and free to use in projects like SourceForge. The database is used to store user information, discussion forums, bug report, etc; basically all the information that is dynamically displayed on the web pages.

Version Control

CVS is used for the project’s source code. The majority of open source project uses CVS [Norin and Stöckel, 1998], since it provides version control and the ability for several developers to work with the same source code simultaneously. Every changes that is committed to the CVS repository is stored. This gives the developers the possibility to backtrack every changes that has been done to the source code since the project was imported into the CVS repository.

Backup

AMANDA\textsuperscript{16} is the backup system used in SourceForge. All site and project data is backed up on tape daily and stored in a fireproof safe. Every week backup tapes are transferred to a company that specializes in storing backup tapes. Due to its capacity the file server is backed up once a month and then incremental every day.

Mailing Lists

Mailing lists are provided with GNU MailMan\textsuperscript{17}. GNU MailMan is a new mailing list system that has gained much popularity the last year. One of the reasons for its success is the easy-to-use web interface. The major mailing list systems like Majordomo have a mail interface that means that the user has to send subscribe commands via mail.

The administration of GNU MailMan mailinglist is also done by a web interface which is both easy for new mailing list administrators and sophisticated for advanced users. [Free Software Foundation, 1999]

\textsuperscript{15}Structured Query Language, the standard for relational databases.

\textsuperscript{16}The AMANDA (Advanced Maryland Automatic Network Disk Archiver, [University of Maryland at College Park, 1999]) project is hosted by SourceForge.

\textsuperscript{17}GNU MailMan is hosted by SourceForge.
Mailing lists are essential to open source projects due to the physical distance that normally separates the developers. [Raymond, 1998]

There are several open source mailing lists systems, but SourceForge has chosen the one with the most “user friendly” interface. Since SourceForge provides a web interface to nearly all its services it’s natural that the mailing lists also has a web interface.

**Web Servers**

Apache [Apache Software Foundation, 1999] and Boa [Doolittle and Nelson, 2000] are used as web servers. Apache has become a de facto standard for the web servers on the Internet. Mainly this is because it’s open source, reliable, fast and scalable. Unlike Apache, Boa\(^\text{18}\) is a single threaded web server.

Since we were unable to obtain a copy of the SourceForge white paper we don’t know where SourceForge uses Boa instead of Apache and why.

**Security**

For secure connections, SourceForge uses SSH [SSH, 2000b] and ModSSL. ModSSL is a module for Apache providing a end-to-end encryption between the web server and the user’s web browser. Encryption is important since all the project administration is done through the web interface. Without encryption, it would be possible for someone to snoop the password to gain full control over the project resources.

SSH provides the same service as rsh\(^\text{19}\) only with encrypted transfer. SSH uses strong encryption to encrypt all communication between two machines in a network, including the username and password. SSH also provides optional compression to reduce the bandwidth needed for the communication. [SSH, 2000a] No organization that really cares about security should accept unencrypted logins, because it is easy to eavesdrop unencrypted passwords over the Internet, which means the security of the whole site could be compromised.

\(^{18}\)The Boa project is hosted by SourceForge.

\(^{19}\)rsh (remote shell) is an application for remote connections to unix machines.
3.6 Interactivity

What parts of SourceForge are interactive, in what way, and to what extent? [Jensen, 1998] defines interactivity in four subgroups (our translation):

Interactivity can be described widely as *a goal for the medium’s potential possibilities of letting the user influence the media distributed communication’s content and/or form.*

**Transmittative interactivity** The user can choose within a continuous stream of information in a media system without a return channel, and therefore without the possibility of user requests (e.g. TV text, near-video-on-demand, be-your-own-editor, multiple channel system)

**Consultative interactivity** The user can choose by request within an existing offer in pre-produced information in a media system with a return channel (e.g. video-on-demand, online information services, CD-ROM encyclopedias)

**Conversational interactivity** The user can create her own information in the media system by input, thereby making them available to other users, whether this happens by a storage medium or in real-time (e.g. video conferencing, newsgroups, e-mail, word processing)

**Registrative interactivity** The media can register information from and therein also adapt to and/or respond to the user’s needs and actions, whether those are explicit choices of communication modes or the system’s built-in ability to automatically adapt by “sensing” (e.g. surveillance systems, intelligent agents, intelligent guides or intelligent user interfaces). Several aspects are within this dimension, given the terms: adaptiveness, responsiveness, recursivity, customization, personalization, intelligent multi media, personal agents, etc. The difference between the consultative and the registrative interactivity is thus a difference between the user’s choice of information content and the media system’s choice of or adaptation of communication modes, that is, the way the communication system works.

We find that Jensen’s definition of interactivity suits our needs well.
3.6. INTERACTIVITY

3.6.1 Transmittative Interactivity

There is little in the way of transmittative interactivity on SourceForge. Arguably, some of the information is "static" as seen from a passive user's point of view (site introduction, documentation, etc), but that information is possible to change through feature requests, bug reports, help and discussion forums about SourceForge itself.

3.6.2 Consultative Interactivity

One of the key features of hyper-linked documents, especially on the WWW, is that the user always has the choice of which information to see, and to a large extent, how. [Nielsen, 1999] stresses the importance of providing information on the web in a manner taking advantage of this unique feature, which printed text doesn't have. Based on this, most of the "static" content on SourceForge offers consultative interactivity to the extent that there are hyper-links.

3.6.3 Conversational and Registrative Interactivity

Most of SourceForge is about conversational and registrative interactivity, processes where the users cooperate, change elements of SourceForge and each other's software, converse via the discussion forums or seek information displayed in a customized manner.

Discussion Forums

The discussion forums provide two forms of conversational interactivity. One is the ability to directly add or modify (perhaps even remove) content on the SourceForge web site, the other is to communicate with other users through the discussion forums.

Search

The search function provides basic conversational interactivity through free text search in limited areas. There are two search options: a Software/Group option (default) and a People option.

The Software/Group option allows searching in the project's title and description, and gives quite good and relevant hits. The word "nttp" gives 9 hits, all of which were relevant. Search time was less than 1 second. The default logic is AND, but it is also possible to use an OR-logic by un-checking the box marked "Require All Words".

38
CHAPTER 3. ANALYSIS AND EVALUATION

The People search functionality lets you search for users by user login ID or real name. It is as fast and accurate as the Software Search. An annoying feature is that when the search is submitted, Software/Group search is again the default.

Code Snippets

In section 3.4.4 we briefly describe the usability questions around code snippets, which we think are adequate in describing parts of the interactivity aspect.

The code snippet feature seems best classified as registrative interactivity, since the underlying system does some automatic extraction of information from the submission process to present the code snippets.

Software Projects

Participating in software projects is probably the most complex interactive feature of SourceForge. Users can start new projects (through a well-defined procedure that includes describing the project with words, licensing and software category) quite easily, thereby adding to the SourceForge web site (projects are granted both "normal" web space in terms of a home page/subsite as well as a general project access page), they can rate the project on a satisfaction scale, monitor changes in projects (on package or file level), contribute to the discussion forums dedicated to the project, track bugs, look through the source, and last but not least participate in the project itself.

Personal Account

A regular user can register with SourceForge and get a personal account, which makes it possible to participate in projects and supply software via SourceForge, and to some extent configure one’s "personal page" (see section 3.4.9). However, the personalization features are directed on organizing development and debugging information on a single page, and leave no room for personal navigation elements or personalization of the rest of the site.
Chapter 4

Improvements

4.1 Introduction

In the prototype, we try to address some of the issues that arose in the initial analysis of SourceForge by suggesting improvements.

Due to time constraints, we have decided not to reuse SourceForge’s complex PHP and MySQL code, but rather develop a very simplified version to illustrate our points (and nothing more). This leaves the prototype as a hardly functional mock-up, as logging in and out doesn’t work, nor does actual update of documents or projects, so the interactivity aspect has hardly been touched.

We’ll be improving SourceForge from the registered user’s point of view, with weight on the developer.
4.2 Main Page

In the improved main page, we have chosen a simpler design than the one in the original SourceForge site (see figures 4.1-4.7, and section A.2.13 for PHP source). We have considered the following points in particular:

4.2.1 Design

- **Screen real estate**
  
  [Nielsen, 1999] We have done little about the screen real estate taken by navigational elements, except by moving all global navigation to the top of the page, leaving room for local navigation on the left.

- **Perceived response time**
  
  [Nielsen, 1999] We have focused on creating simpler HTML, which reduces transmission and rendering time.

- **Style sheets**
  
  SourceForge uses them, but not extensively, therefore it makes it difficult to change the design later. We have employed style sheets to a greater and possibly better degree (see section A.1 for original and new CSS files), which makes site layout and color choice changes much easier [Nielsen, 1999].

Figure 4.1: New Main Page Design, Anonymous User

![New Main Page Design](sourceforge.png)
CHAPTER 4. IMPROVEMENTS

Figure 4.2: New Main Page Design, Anonymous User (2)

Figure 4.3: Simpler Main Page Design, Anonymous User (3)
4.2. MAIN PAGE

Figure 4.4: Simpler Main Page Design, Anonymous User (4)

Figure 4.5: New Main Page Design, Logged In User
CHAPTER 4. IMPROVEMENTS

Figure 4.6: New Main Page Design, Logged In User (2)

Figure 4.7: Simpler Main Page Design, Logged In User (3)
• **Color choice/contrast**  
  [Dix et al., 1993], [Nielsen, 1999] The color scheme has been enhanced slightly for better internal consistency and color contrast in navigation bars (global and local, but not the top navigation strip, where we decided to go for the original color choice — “color contrasts” [Dix et al., 1993].
4.3 Information Architecture

Information in global and local navigation has been grouped thematically in a planned labeling system, as opposed to a random/unplanned system [Rosenfeld and Morville, 1998]. Figure 4.8 shows the information architecture in a graph as [Rosenfeld and Morville, 1998] recommends.

The navigational system looks inconsistent if seen from a high level. Global navigation has different colors for links (white, blue and bold blue), the groups have different arrangements for different purposes, and the link colors don’t change when visited (as they do with embedded links.) However, this is a plus in the way that it helps differentiate the different parts of the navigational elements. [Grudin, 1989] supports the view that too much consistency is bad, and we’ve tried to be consistent only in that we represent the same navigation in the same fashion from page to page, yet the different parts of the navigation system remain reasonably clear.

4.3.1 Search

We have added site-wide search, as we find the current search options too limiting. See figure 4.9.

4.3.2 Navigation, Links and Labeling

We have made a clear distinction between global and local navigation.

The navigation bars have been changed to not self-link back to the current page, and we’ve removed many redundant links to the same places (see figures 4.1-4.7, and source code for the PHP implementation of the global navigation system, sections A.2.5-A.2.6).

The structure has been changed slightly to allow for better navigation.

Navigation labels have been made easier to read and understand, and more in line with what the pages actually describe.

The number of navigation labels has been reduced, so has the complexity (fewer words in the link label), which should make them easier to remember and navigate in ([Dix et al., 1993], “short-term memory”).

The embedded link labels have been made clearer, and now appear in a context (if you’re logged in, you get different links on the main page than when you’re not, for instance — [Rosenfeld and Morville, 1998], “building context”).

With the same modifications as in section 3.3
4.3. INFORMATION ARCHITECTURE

Figure 4.8: Improved Information Architecture
CHAPTER 4. IMPROVEMENTS

Figure 4.9: Simpler Main Page Design, New Search Option (Site Search)

Visited links still don’t have different colors from unvisited links. Although [Nielsen, 1999] likes to have such different colors, they only help the first time you browse the site, and not afterwards (depending slightly on browser configuration). Although it might be a decent helper in navigating, the navigation system should be good enough (and we hope we have achieved that) that this shouldn’t be a major problem.

(In addition to this, there is another point. To allow users to recognize their location on the site, navigation systems should be as consistent as possible. Inline links and navigation links are inherently different, because navigation links are "part of the terrain" a user uses to navigate.)

Global navigation has been placed at the top of each page, where it stays almost no matter where you are in the site structure.
4.4 User’s Personalized Work Environment

The user’s own personalized section as depicted in figure 4.10.

4.4.1 Personal Page

The personal page has been improved with focus on overview and easy of use. We have added personalized navigation, enabling the user to have his favorite project in the local navigation area for quick access. We have also simplified the account maintenance page by putting more functionality available more quickly.

Local navigation

My Projects Here the user’s personal projects are listed. E.g. “Foo”, “GnuBar 1.0”, “Ladida”, “Gruff 2.0”, “Smegg 1.0”, etc. These can be set up in the “Monitoring section”.

User Administration Issues such as “Messages”, “Account Maintenance”, and “Monitoring” go here.
CHAPTER 4. IMPROVEMENTS

Content Area

The content area shows items such as tasks, forums, bug-tracking, monitored file modules, etc. This area is almost identical to the original SourceForge layout.

Account Maintenance

We have placed editing functionality for personal information on this page, instead of linking to other pages. The rest of the page is quite similar to the original SourceForge page.

Monitoring

On this page, monitoring accounts can be added and removed. Monitored forums can be sent directly to the user via email. Projects can either be monitored, put up for the personal navigation bar or both. See figures 4.11, 4.12.
4.4. USER'S PERSONALIZED WORK ENVIRONMENT

Figure 4.12: Monitoring Settings (2)
Chapter 5

Conclusion

SourceForge has gained popularity in the open source movement since the project started in late 1999, and is now of great importance, because of its unique features in providing a gathering place for core developers, advanced and regular users. We believe that SourceForge will be instrumental to the continued success of open source development in the future.

There are many registered users active in many software projects, which is a clear indication that the overall usability of the site is good [Nielsen, 1999], [Nielsen, 2000], yet we found several points to improve, and not only from a usability point of view.

The cluttering of navigational elements in a seemingly unplanned or random manner on SourceForge’s original web pages is a disadvantage for effective use of the site. The messy information architecture makes it difficult to navigate, which again compounds the annoyances in a development process.

In spite of these problems, SourceForge works, with room for improvements. With our efforts, we believe that SourceForge can become even more attractive in use by the open source community.

Our own work was hampered by various problems (as mentioned in chapter 2), and resulted in a continuous lowering of expectations for what we would finally accomplish technically. We set out to make a functional prototype, but ended up with a mock-up. Fortunately, this does not present a problem in the case of making our point about what could be done better with SourceForge, and how.
5.1 Future Work

In light of our analysis, evaluation and improvements, there is still a lot of work to be done. There are several points we should have improved on according to our evaluation, such as the interactivity aspect, especially with relation to the code snippet library. These things need to be addressed in order to make SourceForge as good as we believe it deserves to be.

Our prototype is not compatible with the source code of SourceForge, which means that many of our improvements will have to be reimplemented from scratch to fit in, such as our session (login/logout) handling (which didn’t work). The automated navigation highlighting system is generic enough to be merged into SourceForge with minor modifications.

A graphical redesign of the page layout would have been interesting to do, taking a look at different uses of colors and contrasts to improve on communication with the users and usability.

There is still room for many improvements. We hope to inspire the SourceForge team to incorporate our changes with our help, and to take the step towards further improvements of an already excellent service.
Bibliography


BIBLIOGRAPHY

http://slashdot.org/article.pl?id=00/04/18/1949239&amp;mode=thread.

Multimedia, Hypermedia, Interaktive Medier, pages 125–164. Aalborg
Universitetsforlag.


level 1.
http://www.w3.org/TR/REC-CSS1.

Linux Virtual Server Project.

http://linux.com/about/.


Å gi brukerne maktt gjennom kompetanse og frihet.
http://www.stud.ifi.uio.no/~andreasn/hfag/essay364/.

Simplicity. New Riders Publishing, Indianapolis, Indiana USA.

http://slashdot.org/article.pl?id=00/03/096223&amp;mode=nocomment.

software development methodology. Master’s thesis, Luleå university of
technology.

to open source.


[The \TeX\ Project, 1999] The \TeX\ Project (1999). An Introduction to \LaTeX. http://www.latex-project.org/intro.html.


Appendices
Appendix A

Source Code

This appendix contains the most relevant source code for our prototype (mock-up) of the improved SourceForge site.
A.1 Style Sheets

The style sheets are CSS level 1 ([Lie and Bos, 1996]).
A.1.1 Original Style Sheet

/* SourceForge style rules */
BODY.sf { background-color: #FFFFFF }
A.sf { text-decoration: none; color: #6666FF; }
A.sf:visited { text-decoration: none; color: #6666DD; }
A.sf:link { text-decoration: none; color: #6666DD; }
A.sf:active { text-decoration: none; color: #6666DD; }
A.sf:hover { text-decoration: none; color: #FF6666 }
OL.sf,UL.sf,P.sf,BODY.sf,TD.sf,TR.sf,TH.sf,FORM.sf,
SPAN.sf { font-family: arial,helvetica,sans-serif; color:
    #333333 }
H1.sf,H2.sf,H3.sf,H4.sf,H5.sf,H6.sf { font-family:
    arial,helvetica,sans-serif }
PRE.sf,TT.sf { font-family: courier,sans-serif }
SPAN.sfcenter { text-align: center }
SPAN.sfboxspace { font-size: 2pt; }
A.sfmaintitlebar { color: #FFFFFF; text-decoration: none }
A.sfsortbutton { color: #FFFFFF; text-decoration: underline; }
A.sfmenus { color: #000000; text-decoration: underline; }
A.sftabs { color: #000000; text-decoration: underline; }
SPAN.sfalignright { font-family: arial,helvetica,sans-serif;
    color: #333333; text-align: right }
SPAN.sfmaintitlebar { font-family: arial,helvetica,sans-serif;
    color: #333333; font-size: 10pt; color: #FFFFFF }
SPAN.sftitlebar { font-family: arial,helvetica,sans-serif;color:
    #333333; text-align: center; font-size: 10pt; color:
    #FFFFFF; font-weight: bold }
TD.sffeaturebox { font-size: small; }
/* End SourceForge style rules */
A.1. STYLE SHEETS

A.1.2 Improved Style Sheet

BODY {
  background-color: #FFFFFF;
  margin: 0em;
  font-family: arial, helvetica, sans-serif;
  color: #333333;
  padding: 0;
  margin: 0;
}

/* Common traits */
DL, OL, UL, P, TD, TR, TH, FORM, SPAN {
  font-family: arial, helvetica, sans-serif;
  color: #333333;
}
H1, H2, H3, H4, H5, H6 {
  font-family: arial, helvetica, sans-serif;
}
PRE, TT {
  font-family: courier, monospace;
}
A {
  text-decoration: none;
}
A:active {
  color: #8888FF;
}
A:hover {
  color: #FF6666;
}
A:link {
  color: #555588;
}
A.pageheader:visited {
  color: #555588;
}
A.navglobal:visited {
  color: #555588;
}
APPENDIX A. SOURCE CODE

})

A.navlocal:visited {
  color: #555588;
}

TABLE {
  padding: 0em;
  margin: 0em;
}

TABLE.navtop, TABLE.footer, TABLE.navlocalheader,
TR.navtop, TR.footer, TR.navlocalheader {
  width: 100%;
  background-color: #737b9c;
  font-weight: bold;
  color: #FFFFFF;
}

TABLE.navglobal, TABLE.pageheader, TR.navglobal, TR.pageheader {
  width: 100%;
  vertical-align: middle;
  background-color: #EEEEF8;
}

TR.navlocal {
  vertical-align: top;
}

TR.navlocalheader {
  vertical-align: top;
  background-color: #737b9c;
}

TD.navlocalheader {
  background-color: #737b9c;
  font-weight: bold;
  color: #FFFFFF;
  text-align: center;
}

TD.navlocal {

A-5
A.1. STYLE SHEETS

    background-color: #EFEFE8;
}

TD.searchheader {
    background-color: #73b9c;
    font-weight: bold;
    color: #FFFFFF;
    text-align: center;
}

TD.search {
    font-size: 80%;
    background-color: #EFEFE8;
}

TD.navtop {
    font-weight: bold;
    color: #FFFFFF;
}

TD.navtopleft {
    float: left;
    text-align: left;
    color: #FFFFFF;
    font-weight: bold;
}

TD.navtopright {
    float: right;
    text-align: right;
    color: #FFFFFF;
    font-weight: bold;
}

TD.content {
    vertical-align: top;
    padding-top: 1em;
}

TD.footer {
    color: #FFFFFF;
    font-weight: bold;
font-size: 10pt;
text-align: center;
}

TD.allcontent {
vertical-align: top;
}

SPAN.navtop {

color: #FFFFFF;
}

A.navtop {

color: #FFFFFF;
font-weight: bold;
}

DL.navlocal {

color: #000000;
font-family: arial,helvetica,sans-serif;
}

FORM.search {

font-size: 8pt;
text-align: center;
}
A.2 PHP3 FILES

A.2 PHP3 Files

The PHP3 ([PHP, 2000]) include files form the basic skeleton behind all our pages, as well as providing automatic “fading” of the current page in global and local navigation. In this section, we also present the main page.
A.2.1 Definitions

```php
<?
/**
** def.inc
**
** Some global definitions
*/
$G_USER = ""; /* The user name */
$base_url = "http://tsathoggua.rlyeh.net/~sourceforge/prototype";
    /* Normally: http://sourceforge.net */
$sf_logo_url = $base_url . "/pics/sflogo2-steel.png";
$va_logo_url = $base_url . "/pics/vafllogo.png";
$counter_url = $base_url . "/pics/clear.gif";
$blank_url = $base_url . "/pics/blank.gif";
$right_arrow_url = $base_url . "/pics/right_arrow.gif";
$PHP_suffix = "php3"; /* File name suffix, normally: php */
?>
```
A.2.2 HTML Head

```php
<?
/*
** head.inc
**
** HTML head
*/

$BASE_TITLE = "SourceForge";
$DEFAULT_TITLE = $BASE_TITLE;
$DEFAULT_CSS = "new-SF.css";

function Gen_Head ($title, $author, $description, $keywords, $copyright, $css_doc) {
    global $base_url;
    global $front_page;
    global $DEFAULT_TITLE;
    global $DEFAULT_CSS;
    global $logout;
    global $G_USER;

    if ($logout == "y") {
        user_logout();
    }

    $head = "<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" "http://www.w3.org/TR/REC-html40/loose.dtd">\n
    if (!$title || !is_string ($title)) {
        $title = $DEFAULT_TITLE;
    }

    $head .= "<title>" . $title . " </title>\n
    if (!$css_doc || !is_string ($css_doc)) {
        $css_doc = $DEFAULT_CSS;
    }
```
APPENDIX A. SOURCE CODE

```php
$head .= " <link rel="stylesheet" type="text/css" title="Styles" media="screen" href="/css_doc"/>

if ($author && is_string ($author)) {
    $head .= " <meta name="Author" content="" далее "$author"/>

} 

if ($description && is_string ($description)) {
    $head .= " <meta name="Description" content="" далее "$description"/>

} 

if ($keywords && is_string ($keywords)) {
    $head .= " <meta name="Keywords" content="" далее "$keywords"/>

} 

if ($copyright && is_string ($copyright)) {
    $head .= " <meta name="Copyright" content="" далее "$copyright"/>

} 

$head .= "\n" далее 
    "\</head>\n\n"

return $head;

?>
```
A.2.3 Session Handler

```php
<?
/*
** session.inc
**
** Session control
*/

function session_cookie ($name, $val) {
    setcookie ($name, $val, 0, "/");
}
?>
```
A.2.4 Login Status

```php
<?
/*
** login_status.inc
**
** Login status controller
*/
require ('session.inc');

function user_isloggedin () {
    global $G_USER;
    return ($G_USER);
}

function user_login ($user, $pwd) {
    global $G_USER;
    if ($user && is_string($user) && $pwd && is_string($pwd)) {
        $G_USER = $user;
        setcookie ('SF_LOGGED_IN', "yes", 0, "/");
        return $G_USER;
    } else {
        return 0;
    }
}

function user_logout () {
    global $G_USER;
    unset ($G_USER);
    setcookie ('SF_LOGGED_IN', "no", time()-100000); /* Expire long ago */
}
?>
```
A.2.5 Navigation Class

```php
<?
/*
** navigation.inc
**
** General navigation stuff
*/
class Nav_Item {
    var $doc_name;
    var $url;
    var $label;
    var $desc;

    function Nav_Item ($n, $l, $d, $u) {
        $this->doc_name = $n;
        if (!$u || !is_string ($u)) {
            $u = $n . ".php3";
        }
        $this->url = $u;
        $this->label = $l;
        if ($d && is_string($d)) {
            $this->desc = $d;
        } else {
            $this->desc = "";
        }
    }
}
?>
```
A.2.6 Global Navigation Handler 1

```php
<?
/*
 ** nav_top.inc
 **
 ** Top navigation
 */

/*
 ** function Gen_Nav_Top ()
 **
 ** ARGUMENTS
 **
 ** $nav_item_array (Required)
 **
 ** $css_current_class (Required)
 **
 ** $css_class (Optional)
 */

function Gen_Nav_Top ($nav_item_array,$css_current_class,$css_class) {
    global $doc_name; /* The name of the current document */

    if (!$nav_item_array || !is_array ($nav_item_array)) {
        return;
    }

    $nav_list = "<!-- Beginning of top navigation bar -->\n";
    $nav_list .= "<table>"
    if ($css_class && is_string ($css_class)) {
        $nav_list .= " class="" . $css_class . """
    }
    $nav_list .= " cellspacing="0" width="100\%" border="0">\n"

    $nav_list .= " <tr>"
    if ($css_class && is_string ($css_class)) {
        $nav_list .= " class="" . $css_class . ""
    }
    $nav_list .= "</tr>"

    $nav_list .= " <td>"
    if ($css_class && is_string ($css_class)) {
        $nav_list .= " class="" . $css_class . ""
    }
    $nav_list .= "</td>"
```
$nav_list .= "\n\n\n";

/* Make sure that it prints only'|' between elements. */
$sz = sizeof ($nav_item_array);
$i = 0;
while (list ($key, $value) = each ($nav_item_array)) {
    if (!$value || !is_object ($value)) {
        return;
    }
    if ($value-&gt;doc_name == $doc_name) {
        $nav_list .= "&lt;span";
        if ($css_current_class &amp;&amp; is_string ($css_current_class)) {
            $nav_list .= " class="" . $css_current_class . """;
        }
        $nav_list .= ">" . $value-&gt;label . "&lt;/span&gt;";
    } else {
        $nav_list .= " &lt;a";
        if ($css_class &amp;&amp; is_string ($css_class)) {
            $nav_list .= " class="" . $css_class . """;
        }
        $nav_list .= " href="" . $value-&gt;url . "">" . $value-&gt;label . "&lt;/a&gt;";
    }
    if ($i != ($sz - 1)) {
        $nav_list .= " | ";
    }
    $nav_list .= 

    $nav_list .= " &lt;/td&gt;\n";
$nav_list .= " &lt;td";
if ($css_class &amp;&amp; is_string ($css_class)) {
    $nav_list .= " class="" . $css_class . "right"";
}
$nav_list .= ">\n";
$nav_list .= " &lt;a";
if ($css_class &amp;&amp; is_string ($css_class)) {
    $nav_list .= " class="" . $css_class . """
}
$nav_list .= " href="http://linux.com">linux.com partner&lt;/a&gt;\n";

A-16
$nav_list .= "     </td><br/>
";
$nav_list .= "     </tr><br/>
";
$nav_list .= "</table><br/>
";
$nav_list .= "<!-- End of top navigation bar -->\n"

    return $nav_list;
}
?>
A.2.7 Global Navigation Handler 2

```php
<?
/*
 ** nav_global.inc
 **
 ** Global navigation
 */
function Gen_Nav_Global ($nav_item_array,$css_current_class,$css_class, $n) {
    global $label;

    $doc_name = $label;

    if (!$nav_item_array || !is_array ($nav_item_array)) {
        return;
    }

    $nav_list = "<!-- Beginning of global navigation bar " . $n . " -->\n";
    $sz = sizeof ($nav_item_array);
    $i=0;
    while (list ($key, $value) = each ($nav_item_array)) {
        if (!$value || !is_object ($value)) {
            return;
        }
        if ($value->doc_name == $doc_name) {
            $nav_list .= "<span";
            if ($css_current_class && is_string ($css_current_class)) {
                $nav_list .= " class="" . $css_current_class . """;
            }
        } else {
            $nav_list .= "<a";
            if ($css_class && is_string ($css_class)) {
                $nav_list .= " class="" . $css_class . """;
            }
        }
        $nav_list .= " href="" . $value->url . "">" . $value->label . "</a>";
    } else {
        $nav_list .= "<br>"
    }
    if ($i != ($sz - 1)) {
        $nav_list .= "\n";
    }
}
```
APPENDIX A. SOURCE CODE

```php
}  
$i++;
}

$nav_list .= "\n<!- End of global navigation bar " . $n . " -->\n";

return $nav_list;
}
?>
```
A.2.8 Local Navigation Handler

```php
function Gen_Nav_Local ($nav_header,$nav_item_array,$css_current_class, $css_class,$n) {
    global $label;
    global $right Arrow_url;
    global $G_USER;

    $doc_name = $label;

    if (!$nav_item_array || !is_array ($nav_item_array)) {
        return;
    }

    $nav_list = "<!-- Beginning of local navigation bar " . $n . " -->\n";

    if ($nav_header && is_string ($nav_header)) {
        $nav_list .= "  <tr";
        if ($css_class && is_string ($css_class)) {
            $nav_list .= " class="" . $css_class . "header"";
        }
        $nav_list .= "\n"
    }
    $nav_list .= "  <td";
    if ($css_class && is_string ($css_class)) {
        $nav_list .= " class="" . $css_class . "header"";
    }
    $nav_list .= "\n"
    $nav_list .= "  " . $nav_header . "\n"
    $nav_list .= "    " . $nav_list . "</td>\n"
    $nav_list .= "  " . $nav_list . "</tr>\n"
}

$nav_list .= "  <tr";
if ($css_class && is_string ($css_class)) {
    $nav_list .= " class="" . $css_class . "";
```
APPENDIX A. SOURCE CODE

```php

$nav_list = "">\n";
$nav_list = " <td";
if ($css_class && is_string ($css_class)) {
    $nav_list = " class="" . $css_class . "";"
}
$nav_list = "">\n";

$nav_list = " <dl";
if ($css_class && is_string ($css_class)) {
    $nav_list = " class="" . $css_class . "";"
}
$nav_list = "">\n";

$sz = sizeof ($nav_item_array);
while (list ($key, $value) = each ($nav_item_array)) {
    if (!$value || !is_object ($value)) {
        return;
    }
    $nav_list = " <dt";
    $nav_list = "<img";
    if ($css_class && is_string ($css_class)) {
        $nav_list = " class="" . $css_class . "";"
    }
    $nav_list = "src="" . $right_arrow_url . "" border="0" alt="" >"";
    if ($i != ($sz - 1)) {
        $nav_list = "&nbsp;";
    }
    if ($value->doc_name == $doc_name) {
        $nav_list = "<span";
        if ($css_current_class && is_string ($css_current_class)) {
            $nav_list = " class="" . $css_current_class . "";"
        }
        $nav_list = "">" . $value->label . "</span>";
    } else {
        $nav_list = "<a";
        if ($css_class && is_string ($css_class)) {
            $nav_list = " class="" . $css_class . "";"
        }
        $nav_list = " href="" . $value->url . "">" ;
        $nav_list = $value->label . "</a>";
```
$nav_list .= "\n"
}

$nav_list .= "\n";
$nav_list .= "\n";
$nav_list .= "\n";
$nav_list .= "<!-- End of local navigation bar " . $n . " -->\n";

return $nav_list;
}

function Gen_Search_Local ($search_header,$search_action,$search_method, $css_class) {
if (!$search_action || !is_string ($search_action)) {
    $search_action = "/search/";
}
if (!$search_method || !is_string ($search_method)) {
    $search_method = "post";
}

$search_list = "<!-- Beginning of search box -->\n";
if ($search_header && is_string ($search_header)) {
    $search_list .= "<tr";
    if ($css_class && is_string ($css_class)) {
        $search_list .= " class="" . $css_class . "header"";
    }
    $search_list .= "\n";
    $search_list .= "<td";
    if ($css_class && is_string ($css_class)) {
        $search_list .= " class="" . $css_class . "header"";
    }
    $search_list .= "\n";
    $search_list .= "" . $search_header . "\n";
    $search_list .= "\n";
    $search_list .= "</td>\n";
    $search_list .= "</tr>\n";
}

$search_list .= "<tr";
if ($css_class && is_string ($css_class)) {
    $search_list .= " class="" . $css_class . "";
}
```php
$search_list .= "\n";
$search_list .= " <td>
if ($css_class && is_string ($css_class)) {
    $search_list .= " class="" . $css_class . ""
} $search_list .= "\n";

$search_list .= " <font size="-2">\n";
$search_list .= " <form>
if ($css_class && is_string ($css_class)) {
    $search_list .= " class="" . $css_class . ""
} $search_list .= " action="" . $search_action . ""
$search_list .= " method="" . $search_method . "\n"

$search_list .= " <select name="type_of_search">\n";
$search_list .= " <option value="soft">"
    "Software/Group</option>
$search_list .= " <option value="people">"
    "Developers</option>
$search_list .= " <option value="site">"
    "Entire SF</option>
$search_list .= " </select><br>
$search_list .= " <input type="checkbox"
    " name="exact" value="1" checked>
    "Require all words\n"
$search_list .= " <input type="hidden"
    " name="forum_id" value="\n"
$search_list .= " <input type="hidden"
    " name="is_bug_page" value="\n"
$search_list .= " <input type="hidden"
    " name="group_id" value="\n"
$search_list .= " <input type="text"
    " name="words" value="\n" size="15"><br>\n"
$search_list .= " <input type="submit"
    " name="Search" value="Search">

</form>
$search_list .= " <td>
$search_list .= " </tr>
```

A-23
$search_list .= "<!-- End of search box -->\n";
    return $search_list;
} 
?>
A.2.9 Page Header

```php
<?
/*
** header.inc
**
** Page header, including global navigation
*/
require ('login_status.inc'); /* Login status keeper */
require ('nav_top.inc'); /* Top navigation ("About", "Partners", etc) */
require ('nav_global.inc'); /* Global navigation */

function Gen_Header ($css_class,$sf_compat) {
    global $label;
    global $base_url;
    global $sf_logo_url;
    global $va_logo_url;
    global $PHP_suffix;
    global $G_USER;
    if (user_isloggedin() || $G_USER) {
        $extra = "?G_USER=$G_USER";
    } else {
        $extra = "";
    }
    $header = "<body";
    if ($sf_compat) {
        $header .= " topmargin="0" bottommargin="0" leftmargin="0""
            . " rightmargin="0" marginheight="0" marginwidth="0"";
    }
    $header .= "">n";

    /* Top navigational items */
    $a = new Nav_Item ("about",
        "About",
        "What is SourceForge?",
        $base_url . "/about." . $PHP_suffix . $extra);

    $b = new Nav_Item ("partners",
        "Partners",
        "Partner organizations",
        $base_url . "/partners." . $PHP_suffix . $extra);
```
$c = new Nav_Item("contact",
    "Contact Us",
    "Contact information for SourceForge",
    $base_url . "/contact" . $PHP_suffix . $extra);

$nav_top = array($a,$b,$c);

$header .= Gen_Nav_Top($nav_top,"navtop_current","navtop");

/* Global navigational items */
$header .= "<table";
if ($css_class && is_string($css_class)) {
    $header .= " class="" . $css_class . "";
}
$header .= " width="100\%" border="0" cellspacing="0">\n";
$header .= " <tr";
if ($css_class && is_string($css_class)) {
    $header .= " class="" . $css_class . "";
}
$header .= " >\n";
$header .= " <td";
if ($css_class && is_string($css_class)) {
    $header .= " class="" . $css_class . "";
}
$header .= " rowspan="2">\n";
if ($label != "home") {
    /* Link to home page */
    $header .= " <a";
    if ($css_class && is_string($css_class)) {
        $header .= " class="" . $css_class . "";
    }
    $header .= " href="" . $base_url . $extra . "/" >\n";
}
/* Main logo */
$header .= " <img";
if ($css_class && is_string($css_class)) {
    $header .= " class="" . $css_class . "";
}
$header .= " src="" . $sf_logo_url . "";
$header .= " alt="SourceForge" border="0" width="143" height="70">\n";
APPENDIX A. SOURCE CODE

```php
if ($label != "home") {
    $header .= "   </a>
"
}
$header .= "   </td>
"

/* User status */
$header .= "   <td>
if ($css_class && is_string ($css_class)) {
    $header .= "    class="" . $css_class . ""
}
$header .= ">
"
if (user_isloggedin()) {
    $header .= "    <strong>Status: Logged in as " . $G_USER . "</strong>
"
} else {
    $header .= "    <strong>Status: Not <a>
if ($css_class && is_string ($css_class)) {
    $header .= "    class="" . $css_class . ""
}
$header .= " href="" . $base_url;
$header .= "</account/login." . $PHP_suffix . "">logged in"
    . "</a></strong>
"
}
$header .= "   </td>
"

/* Login/new user or logout/homepage shortcut */
$header .= "   <td>
if ($css_class && is_string ($css_class)) {
    $header .= "    class="" . $css_class . ""
}
$header .= ">
"
if ($css_class && is_string ($css_class)) {
    $header .= "    <a>
if ($css_class && is_string ($css_class)) {
    $header .= "    class="" . $css_class . ""
}
$header .= " href="" . $base_url;
if (user_isloggedin()) {
    $header .= "</index." . $PHP_suffix . "?logout=y">"
    $header .= "<strong>Logout</strong>
"
} else {
    $header .= "</account/login." . $PHP_suffix . "">
```

A-27
\$header .= "<strong>Login</strong>";
}
\$header .= "</a>");
if (\$label != "user") {
\$header .= "&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&n...
```php
$header = " alt="VA Linux Systems" align="right" border="0"
 . " width="117" height="70">

if ($css_class && is_string ($css_class)) {
    $header .= " class="" . $css_class . "";
}
$header .= ">

if ($css_class && is_string ($css_class)) {
    $header .= " class="" . $css_class . "";
}
$header .= ">

/* Normal global navigation elements */
$header = " <tr">
$a = new Nav_Item ("softwaremap",
 "Software Projects",
 "Software Projects hosted by SourceForge",
 $base_url . "/projects/softwaremap/" . $extra);

$b = new Nav_Item ("new",
 "Latest Project Releases",
 "Most recent releases of current projects",
 $base_url . "/projects/latest/" . $extra);

$c = new Nav_Item ("snippet",
 "Code Snippets",
 "Code snippet library, smaller programs that"
 . " aren’t part of a project",
 $base_url . "/snippet/" . $extra);

$nav_global = array ($a,$b,$c);

/* Do the magic generation of bar #1 */
$header .= Gen_Nav_Global ($nav_global,"navglobal_current",
 "navglobal",1);
```
$header .= "    </td>
";
$header .= "    <td>
";
if ($css_class &\& is_string ($css_class)) {
    $header .= "   class="" . $css_class . ""
";
}
$header .= ">
";

/* Build elements, bar #2 */
$a = new Nav_Item ("docs",
         "Docs, Help &amp; FAQ",
         "SourceForge Documentation, help pages and FAQ",
         $base_url . "/docs/" . $extra);

if ($extra != ") {
    $b = new Nav_Item ("discussion",
         "Discussion Forum",
         "Discussion forum for discussions about SourceForge itself",
         $base_url . "/forum/forum." . $PHP_suffix . $extra
             . "&forum_id=2") ;
} else {
    $b = new Nav_Item ("discussion",
         "Discussion Forum",
         "Discussion forum for discussions about SourceForge itself",
         $base_url . "/forum/forum." . $PHP_suffix . "?forum_id=2") ;
}
$c = new Nav_Item ("mirrors",
         "Other Project Mirrors",
         "Mirrors of other projects",
         $base_url . "/mirrors/" . $extra);

$nav_global = array ($a,$b,$c);

/* Do the magic generation of bar #2 */
$header .= Gen_Nav_Global ($nav_global,"navglobal_current","navglobal",2);

$header .= "    <img"
if ($css_class &\& is_string ($css_class)) {
    $header .= "   class="" . $css_class . ""
";
}
$header .= " src="" . $counter_url . ""
";
```php
$header .= " width="1" height="1" alt="Counter">\n";
$header .= "   </td>\n";
$header .= "   </tr>\n";
$header .= "</table>\n";

return $header;
}
?>
```
A.2. PHP3 FILES

A.2.10 Content Base

```php
<?
/*
** content.inc
**
** Basis for pages
*/
require ('nav_local.inc'); /* Local navigation */

function Gen_Content_Start ($css_class) {
    $base .= "<table";
    if ($css_class && is_string ($css_class)) {
        $base .= " class=\"" . $css_class . "\"";
    }
    $base .= " width="100\%" border="0" cellspacing="0" cellspacing="0" cellpadding="0" >\n";
    $base .= " <tr";
    if ($css_class && is_string ($css_class)) {
        $base .= " class=\"" . $css_class . "\"";
    }
    $base .= " >\n";
    $base .= "<!-- Beginning of local navigation -->\n";
    $base .= " <td";
    if ($css_class && is_string ($css_class)) {
        $base .= " class=\"" . $css_class . "\"";
    }
    $base .= " width="19\%" >\n";
    $base .= " </table";
    if ($css_class && is_string ($css_class)) {
        $base .= " class=\"" . $css_class . "\"";
    }
    $base .= " width="100\%" border="0" cellspacing="0" "
        . "cellpadding="0" >\n";
        
    return $base;
}

function Gen_Content_Middle ($css_class) {
    global $url_blank;

    $base = " </table>\n";
```
APPENDIX A. SOURCE CODE

```plaintext
$base .= "  <td>
$base .= "<!-- End of local navigation -->
$base .= "  
if ($css_class && is_string ($css_class)) {
  $base .= " class=" . $css_class . ""
}
$base .= " width="9" bgcolor="#FFFFFF"/>
$base .= "  
  <img src="/" $url_blank . "" alt="" height="1" width="9" border="0">
$base .= "  <td>
$base .= "<!-- Beginning of content -->
$base .= "  
if ($css_class && is_string ($css_class)) {
  $base .= " class=" . $css_class . ""
}
$base .= " width="80%"/>

return $base;
}

function Gen_Content_End () {

  $base = "  
  . 
  . "</table>
  . "<!-- End of content -->

return $base;
}
?>
```
A.2. PHP3 FILES

A.2.11 Page Footer

```php
<?
/*
** footer.inc
**
** Page footer (includes HTML end tags)
*/
function Gen_Footer ($css_class) {
    $footer = "<!-- Beginning of footer -->\n";
    $footer .= "<table";
    if ($css_class && is_string ($css_class)) {
        $footer .= " class="" . $css_class . "";"
    }
    $footer .= " width="100\%" border="0" cellspacing="0"">\n";
    $footer .= " <tr";
    if ($css_class && is_string ($css_class)) {
        $footer .= " class="" . $css_class . "";"
    }
    $footer .= ">\n";
    $footer .= " <td";
    if ($css_class && is_string ($css_class)) {
        $footer .= " class="" . $css_class . "";"
    }
    $footer .= ">\n";
    $footer .= " </td>\n" . " </tr>\n" . " </table>\n" . " </body>\n" . " </html>\n";

    return $footer;
}
?>
```
A.2.12 Standard Page Base

```php
<?
require ('defs.inc'); /* Some necessary definitions */
require ('head.inc'); /* Basic HTML head */
require ('navigation.inc');
    /* Basic navigation class, needed for everything */
require ('header.inc');
    /* Page header, includes global navigation */
require ('content.inc'); /* Content generation stuff */
require ('footer.inc'); /* Footer */
?>
```
A.2. PHP3 FILES

A.2.13 Main Page

<?
/*
** Source Forge improvement suggestion -- not fully functional prototype */

require ('std.inc'); /* Standard inclusion stuff for all normal pages */

$label = "home";

print Gen_Head ("SourceForge","Jan Ingvoldstad",
                "SourceForge - A Community Site for Open Source Development",
                "SF, SourceForge","Jan Ingvoldstad","new-SF.css");
print Gen_Header ("pageheader",0);
print Gen_Content_Start ("allcontent");

/* Local navigation */
$a = new Nav_Item ("top_projects",
                  "Most popular",
                  "The most popular projects",
                  $base_url . "/projects/top." . $PHP_suffix);
$b = new Nav_Item ("newest_projects",
                  "Newest",
                  "The most recently started projects",
                  $base_url . "/projects/newest." . $PHP_suffix);
$c = new Nav_Item ("mostactive_projects",
                  "Most Active",
                  "The most active projects",
                  $base_url . "/projects/mostactive." . $PHP_suffix);
$d = new Nav_Item ("all_projects",
                  "All",
                  "All projects hosted by SourceForge",
                  $base_url . "/search/fullprojectlist." . $PHP_suffix);
$e = new Nav_Item ("sf_feature_request",
                  "Request Feature",
                  "Forum for requesting new features for SourceForge",
                  $base_url . "/forum/forum." . $PHP_suffix
                  . "?forum_id=4");
$f = new Nav_Item ("sf_bugs",
                  "Report Bug",
"Report a bug with SourceForge",
$base_url . "/bugs/?group_id=1";
$g = new Nav_Item ("sf_patches",
"Submit Patch",
"Submit a patch for SourceForge",
$base_url . "/patch/?group_id=1");
$h = new Nav_Item ("sf_support",
"Support",
"Request support for SourceForge",
$base_url . "/support/?group_id=1");
A.2. PHP3 FILES

<?

<TABLE width=100% cellspacing=0 cellpadding=0 border=0>
  <TR>
    <TD width="65%" VALIGN="TOP">
      <TABLE cellspacing=0 cellpadding=1 width=100% border=0 bgcolor=737b9c align="center">
        <TD colspan=2><SPAN class=titlebar>Latest News</SPAN></TD>
      </TABLE>
      <UL>
        <LI><A HREF="/forum/forum.php?forum_id=15619">OpenQueue Is Now Ready</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">mattj - 04/27/00 14:49</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=15630">Many projects have yet to be completed</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">dbressler - 04/27/00 14:41</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=15506">GNU/Hurd Documentation Project</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">jbailey999 - 04/26/00 17:41</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=15433">BB's!---'-- Database Desk</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">pberger - 04/26/00 08:35</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=15381">Generic Document Distributor</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">ajangel - 04/25/00 14:56</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=15358">Big Iron Kicks Off</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">kudzu - 04/25/00 14:56</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=15225">Next Generation Open Interoperability</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">ashvil - 04/24/00 23:26</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=15054">chessd Needs Volunteers for the Future</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">bugg - 04/24/00 09:30</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=15042">Volunteers needed for the Future</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">sambc - 04/24/00 09:18</FONT>
        <LI><A HREF="/forum/forum.php?forum_id=14992">LibTrace looking for C++ developers</A>
            <BR>&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <FONT SIZE="-1">Pauldoo - 04/22/00 10:00</FONT>
      </UL>
  </TD>
</TR>
</TABLE>
SourceForge is a free service to developers offering easy access to the best in CVS, mailing lists, bug tracking, message boards/forums, task management, site hosting, permanent file archival, full backups, and total web-based administration.

In order to get the most out of SourceForge, you need to register as a site user. This will allow you to participate fully in all we have to offer. You may of course browse the site without registering, but will not have access to participate fully.

Set Up Your Own Project

<?
print "<p><a href="/" . $base_url . "/account/register." . $PHP_suffix . ">
Register</a> as a site user, then ";
print "<a href="/" . $base_url . "/account/login." . $PHP_suffix . ">
";
print "Login</a> and finally, ";
} else {
    print "<a href="\" . $base_url . "/register/\">";
}
print "Register Your Project";
if (user_isloggedin()) {
    print "</a>";
}
print ".\n";
print "</p>\n";
?>
<p>
Thanks... and enjoy the site.
</p>
</TD>
<TD>&nbsp;</TD>

<TD width="35%" VALIGN="TOP">
<TABLE cellspacing="0" cellpadding="1" width="100%" border="0" bgcolor="#737b9c">
<TD colspan=2>Your webserver: <B>web1</B></TD>
<TD>
Hosted Projects: <B>3965</B> <A href="/"ié</A>
</TD>
</TR>
</TABLE>
</TD>

<TR BGCOLOR="#737b9c" align="center">
<TD colspan=2><SPAN class=titlebar>Top Project Downloads</SPAN></TD>
</TR>

<TD align=left bgcolor="#F6F6F6">
<TD colspan=2><B>Past 7 days: </B></TD>
<br>
<A href="/project/?group_id=254">Licq</A> (8274)
<br>
<A href="/project/?group_id=978">The Freenet Project</A> (7282)
<br>
<A href="/project/?group_id=3">Mesa3D</A> (7232)
<br>
<A href="/project/?group_id=1897">Aladdin Ghostscript</A> (3743)
<br>
<A href="/project/?group_id=2619">KXicq</A> (2397)
<br>
<A href="/project/?group_id=588">jEdit</A> (2054)
<br>
<A href="/project/?group_id=2479">DooM Legacy</A> (1981)
<br>
<A href="/project/?group_id=2712">EtherApe</A> (1962)
<br>
<A href="/project/?group_id=3570">Halflife Admin Mod</A> (1836)
<br>
<A href="/project/?group_id=518">GNOME Napster Client</A> (1424)
<br>
<P align="center"><A href="/top/">[More Top Projects]</A>
</P>

</TD>
### APPENDIX A. SOURCE CODE

```html
</TR>
<TR BGCOLOR="737b9c" align="center">
<TD colspan=2><SPAN class=titlebar>Newest Projects</SPAN></TD>
</TR>

<TR align=left bgcolor="#F6F6F6">
<TD colspan=2><A href="/project/?group_id=5067">Linux Home Firewall</A> (04/27/2000)<BR>
<A href="/project/?group_id=5064">VisKProg</A> (04/27/2000)<BR>
<A href="/project/?group_id=5062">XRazzle!</A> (04/26/2000)<BR>
<A href="/project/?group_id=5061">501k Analysis Software</A> (04/26/2000)<BR>
<A href="/project/?group_id=5060">Warped Reality</A> (04/26/2000)<BR>
<A href="/project/?group_id=5058">IRCStat Network Statistics Daemon</A> (04/26/2000)<BR>
<A href="/project/?group_id=5057">C++ class library for manipulating audio</A> (04/26/2000)<BR>
<A href="/project/?group_id=5055">FuseObjects</A> (04/26/2000)<BR>
<A href="/project/?group_id=5054">dbishell</A> (04/26/2000)<BR>
<A href="/project/?group_id=5053">Object Persistence Layer</A> (04/26/2000)<BR>
</TD>
</TR>

<TR BGCOLOR="737b9c" align="center">
<TD colspan=2><SPAN class=titlebar>Most Active Projects</SPAN></TD>
</TR>

<TR align=left bgcolor="#F6F6F6">
<TD colspan=2><B>( 99.01075 )</B> <A HREF="/project/?group_id=1">SourceForge</A><BR>
</TD>
</TR>

</TABLE>

print Gen_Footer ("footer");

?>
```
A.2. PHP3 FILES
Appendix B

SourceForge Hardware

Load Balancing Firewall Server
PII 350
128MB RAM
4.5GB U2W SCSI HDD

SourceForge Web Server 1
Dual PIII 600
512MB RAM
18GB U2W SCSI HDD

SourceForge Web Server 2
Dual PIII 600
512MB RAM
18GB U2W SCSI HDD

SourceForge Database Server
Quad PII-Xeon 400
2GB RAM
75GB RAID-5

SourceForge File Server
Quad PII-Xeon 400
2GB RAM
850GB on 5 Mylex ExtremeRaid Controllers

SourceForge Mail/DNS Server
Dual PII-350
512MB RAM
54GB U2W SCSI HDD

Geocrawler Mail Archiver
Dual PIII-500
1GB RAM
154GB RAID-5

Sitewide Backup Server
PII-350
128MB RAM
18GB U2W SCSI HDD
5 Seagate 25GB Native AIT Tape Drives

Project CVS Server
Dual PIII-600
1GB RAM
75GB RAID-5

Project Web Server
Dual PIII-600
512MB RAM
2 18GB U2W SCSI HDD

Project Shell Server
Dual PIII-500
512MB RAM
2 18GB U2W SCSI HDD

Project Database Server
Dual PIII-Xeon 500
1GB RAM
35GB RAID-5