

安全で簡単な Wiki ソフトの作成について

國松 早紀

目 次

1	はじめに	1
1	テーマと動機	1
2	Wiki とは	2
3	それぞれの Wiki ソフトについて	5
4	誰のために作ったか	6
2	ソフトについて	7
1	ソフトの構想	7
2	このソフトでできること	8
3	ファイルの保存	8
4	ゴミ箱の機能	9
5	管理者メニュー	9
6	セキュリティについて	10
7	require の設定	13
8	Wiki 構文について	14
9	デザインの設定	14
3	アンケート	15
1	アンケートの結果	15
2	アンケートの評価をふまえて	17
4	自己評価	19
1	良かった点	19
2	悪かった点	20
5	おわりに	20

1 はじめに

私は、今回の卒業論文のための制作物として、セキュリティ面での安全を確保できて、データベースを使用しない、シンプルで使いやすい Wiki ソフトを作成した。

(1) テーマと動機

私が、「安全で簡単な Wiki ソフトの作成」を卒業制作、及び卒業論文のテーマにした理由としては、今まで私の所属している福田ゼミで使っていた Wiki には様々な問題がある、と先生がおっしゃっていたので、その問題を改善できるような Wiki ソフトを作成するといったテーマに決めた。

現在ある Wiki ソフトではデザインが見つらかったり、荒らし行為による被害を受けたり、何日か書き込みがないと広告が表示されてしまったり、ファイルを自分の手元で管理できなかつたりと、いろいろな不安点がある。

そこで、初心者でも簡単に、セキュリティ面でも安全に Wiki を作成できるソフトが作れないかと考えた。福田ゼミに入る際に、先生と面談をし、初めに聞いた言葉が「人の役に立つものを作る」といったことであつたので、その福田ゼミで役に立つ Wiki ソフトを作りたいと考えたのだ。

このゼミで初めに利用していた Wiki は、「PukiWiki」である。これはデータベースを使わずにファイルを保存できるので安心して利用していたが、海外の業者からのスパム書き込みがとてつもなく多く、ついには開発側が新しいものを配布してくれなくなってしまった。スパムだらけで、ゼミの Wiki サイトを PukiWiki でこのまま続けていくのは困難

だと考え、今度は FC2Wiki をゼミの Wiki サイトとしてとして利用することにした。

PukiWiki の後に利用していた FC2Wiki は、海外の業者からのスパムなども書き込まれないため、しばらく使われていたが、ファイルがデータベースに保存されるため、手元にファイルが残らない。もし何らかの不具合でファイルが消えてしまうと、どこにもバックアップがなく、せっかく作ったものもすべて見られなくなってしまうので、これからずっとこの Wiki を使っていくとなると、不安も残る。

また、デザイン面でも、数個用意されたデザインのテンプレートの中から選ぶことしかできず、なかなか自分の好きなデザインにできないので、自由度が低いといった問題もある。

もう一つ、Wikipedia など使われている MediaWiki というソフトも便利だが、Wordpress⁽¹⁾よりも大きなソフトなのでメンテナンス作業に大きく時間をとられてしまう。少し複雑な作業などもしなくてはならず、ゼミ内での情報交換のためのツールであるはずが、そのツールを使うために時間を取られてしまう可能性も出てくる。

そこで、ゼミの先生や生徒の間で、授業で扱う情報の交換やコミュニケーションをとることが簡単にできて、ファイルも手元にのこるような、安心して自由な Wiki ソフトを作成したいと考え、私の卒業論文のテーマは「安全で簡単な Wiki ソフトの作成について」となった。

(2) Wiki とは

Wiki とは、Web ブラウザを利用し、複数の人数で Web サイトを作成していく Web コンテンツマネジメントシステム (WCMS) のことである。類似したものとしては、ブログやホームページなどがある。これ

は、1人の管理者が Web サイトを管理し、閲覧者は管理者が作成したページを見ることができて、それに対してコメントを書き込むことができるシステムになっている。

Wiki では、管理者や閲覧者といったことは関係なく、誰でもページを新しく作成したり、現在あるページに加筆修正したり、削除したりすることもできる。メニューの編集をすることができるものもある。もちろん、パスワードを設定してページやメニュー編集ができる人を限定することもできる。

HTML がわからない人でも見やすいページを作成できるように、文字の装飾や、画像の挿入、内部・外部リンクの挿入、箇条書きや表を挿入することが可能な、「Wiki 構文」という特殊な整形書式が定められている。それぞれの Wiki によって Wiki 構文は異なっていることもある。

Wiki を利用したインターネット上の Web サイトで、最も有名なものでは Wikipedia がある。アカウントを作成すれば誰でも百科事典の新しいページを作成したり、誰かが作った項目の加筆修正を行ったりすることができる。また、悪質な荒らし行為や犯罪予告が書き込まれると投稿ブロックを行うなど、荒らしへの対策もとられている。

電子掲示板 (BBS) に近いものにも思えるが、Wiki は誰かが書き込んだ内容の編集や削除ができるので、BBS よりも、何人かが一つの Web サイトを作っていく、といったイメージのものである。誰でも簡単に作成できるので、趣味のコミュニティの情報交換に使われたりメモ帳代わりに利用されたりすることもある。

日本にあるもので名前の知られている Wiki は、@ Wiki⁽²⁾や前述の FC2 Wiki⁽³⁾、PukiWiki⁽⁴⁾、また、seesaa Wiki⁽⁵⁾や、MediaWiki⁽⁶⁾などがある。それぞれの特徴については後述する。

(i) Wiki の利点

Wiki を利用する利点としては、前述の通り、何人かで一つの Web サイトを編集できるということであると言える。一般のホームページやブログ、多くの Web サイトは、基本的に作成者以外の人々がコンテンツを変更をすることはできない。誰か一人（または一つのアカウント）しか管理者がおらず、複数人でその Web サイトを管理しようとする、一人に編集してもらうか、アカウントに交互にログインして編集するといった形になってしまう。

しかし、Wiki を利用すれば、パスワードを設定して何人かだけが Web サイトを編集できる方法がとれる。パスワードをあえて設定しなければ、インターネットを利用できる世界中の人々すべてが Web サイトを編集することができるようになり、前述した Wikipedia のような大きなオンライン百科事典を作成することもできる。

Wikipedia は現在、日本語版のみでも 20 万語以上の単語のページがあり、常にたくさんの人の手で思い思いに解説を書き込んだり、修正したり、編集されて、日々進化している。

(ii) Wiki の問題点

一方、Wiki の問題点としては、たくさんある Wiki 構文を細かく記憶しなくてはならないという点が挙がってくる。これは Wiki によって異なり、いくつかの異なる Wiki を利用するとなるときちゃんと使い分けなくてはならない。

本来は、Wiki 構文は HTML などの文法がわからない人でも Web ページが利用できるように設定されたもので、Wiki を使う上での利点と言えた機能だった。しかし、最近では WYSIWYG エディタのようなものも登場してきた。これは、利用者がボタン一つで簡単に HTML

を入力できるエディタであり、様々な種類が次々と公開されている。WYSIWYG エディタを使えば、ワードプロセッサのように Web ページを作成することができるので、「Wiki 構文が利用できる」という便利だった点が、時代が変わるにつれてだんだんと問題点になってきている。

また、データベースを利用している Wiki では、それを利用して作成した Web ページがデータベースに登録される。そしてそのデータベースから閲覧、編集、削除を行うので、データベースに不具合が生じたり、サービスが何らかの理由で突然終了してしまうと、データが自分の手元に残らない。今まで作成してきた Web サイトが水の泡となってしまう可能性もゼロとは言えないというところも、Wiki の問題点だといえる。

(3) それぞれの Wiki ソフトについて

(i) MediaWiki

Wikipedia などで行われているフリーな Wiki ソフトで、1日に何百万人が訪れる Web サイトでも運用ができるような作りになっている。MySQL データベース⁽⁷⁾にデータを保存し、それを PHP で取り出すことができる。しかし、FTP や SQL などの設定があり、初心者が扱うには、最初の設定が少し困難なソフトである。

(ii) レンタル Wiki

レンタル Wiki は、前述した FC2Wiki や、seesaa Wiki、@ Wiki など、それぞれの会社が運営する様々なサービスの中の一つである。メールアドレスなどを用いてアカウントを作成すると、その会社の提供する様々なサービス (Wiki 以外にも、ホームページやブログなど) を利用できるようになる。一般的にはこれらを利用して作成した Wiki の URL

は、サブドメイン形式とサブディレクトリ形式のいずれかとなる。

サブドメイン形式では、FC2Wiki の場合 URL は「http:// (自由に決められる部分) .wiki.fc2.com/」サブディレクトリ形式は、seesaa Wiki の場合、URL は「http://seesaawiki.jp/ (自由に決められる部分) /」となる。

レンタル Wiki を利用すると、サービスのトップページに人気 Wiki ランキングや、最近更新された Wiki ページが表示されることもある。他の Wiki から引っ越しといった形をとれるものや、バックアップ機能を備えたものもある。各々で禁止ワードを設定したり、特定の IP アドレスやホスト名からの投稿をブロックすることができるものもある。

(4) 誰のために作ったか

この Wiki ソフトは、アカウント登録や面倒な手続きをしたくなかったり、安全で、気軽に Wiki が利用したいという人のために作成した。また、福田ゼミで現在使っている「FC2Wiki」に変わって、CSS などもデザインも自由に書き換えられる Wiki として使えるものにする 것을目標として定めて作成した。

テーマを決めた後に、現在ある Wiki ソフトについて調べてみた。すると、アカウント登録を先に済ませてからやっと Web サイトの作成に手をつけられる形になっているものがあったり、ページの作成ページに機能やボタンがたくさんありすぎてわかりにくかったり、英語の説明書きを読んでインストールしなければならなかったりと、Wiki を利用したことがない初心者にとってはなかなか手を出しにくいのではないかと考えた。

私も、過去にレンタルサーバーでブログやホームページを作成してお

り、Wikipedia もたびたび調べ物のために利用していた。しかし、人文情報学科でいろいろなことを学ぶまで Wiki というものの存在や利用法をほとんど知らなかった。そして、Wiki のことを知った後も、Wiki ソフトをインストールする際に、説明が英語のものもあつたり、手順をいくつか踏まなくてはならなかったりと何度もつまづいてしまった。これでは、Wiki ソフトを利用する前にインストールの時点であきらめてしまう人もいるかもしれない。

また、試しに Wiki を利用してみると、新しい Web ページを作成する画面に、たくさん便利な機能が使えるボタンが設置されていた。しかし、なかなかそのボタンの意味が全部わからず、ごちゃごちゃとして見えるため、使い慣れるまでは使いづらく感じるかもしれない。

そこで、専門的な知識が全くなく、Wiki のことが何もわからない初心者でも、ぱっと画面を見てすぐ直感で書き込みやメモなどができ、簡単に Web サイトが作成していけるようなシンプルな Wiki ソフトを作成した。

2 ソフトについて

(1) ソフトの構想

このソフトを制作する上で、「パスワード付きにして安全なものにする」「デザインが自分で自由に設定できるものにする」「ファイルが手元に残るようにする」と、この三点を目標として設定して作成する。今後のゼミ生が Wiki を使用するのが初めてであっても、使用法に悩むことなく使えるよう、シンプルかつわかりやすいデザインにする。

Wiki 構文についても、一般的によく使われている記法を調べて案を考え、難しい記号は使わずにページを装飾できるつくりにする。ソフトの

内容については後述する。

(2) このソフトでできること

このソフトを用いてできることとしては、ユーザーモード (図 1 参照) では、

- ・ ページの閲覧 (図 2 参照)
- ・ ページ名を一覧表示する (図 3 参照)

と、この二つの項目であり、ログイン画面 (図 4 参照) からログインし
てからの管理者モード (図 5 参照) ではこの二つに加えて、

- ・ 新規ページの作成 (図 6 参照)
- ・ ページの編集 (図 7 参照)
- ・ ページをゴミ箱に移す
- ・ ゴミ箱に入っているページを完全に削除する (図 8 参照)
- ・ ゴミ箱に入っているページをゴミ箱から出し、ページ一覧に復活させる
- ・ 左側メニューの編集 (図 9 参照)

ということを行うことができる。

(3) ファイルの保存

まず、Wiki ソフトを作る上でファイルをどこに保存するかということ
を考えた。データベースを使うと、そのデータベースが何らかの理由
で使えなくなった場合、ファイルが永久的に取り出せなくなってしまう
ので、手元のフォルダにどんとどんとたまっていくといった形になるよう
にした。ファイルの名前は、その Web ページのタイトルになるように
した。

具体的には、新しいページを作成し、タイトルと本文をテキストボックスに書き込んだあと、保存するプログラムに`$file_name = "kiji/$title.txt"`と書き込むと、kiji という名前のフォルダに\$title に代入された、先ほどテキストボックスに書き込んだページのタイトルが入り、.txt がついた状態で保存される。

(4) ゴミ箱の機能

ページを削除したい場合は、そのページを表示させると見える「削除」というボタンを押せば、「このページをゴミ箱に移しますか?」というポップアップがでてくる (図 10 参照)。「OK」ボタンを押すと、そのページがゴミ箱というフォルダに移動する。ゴミ箱にあるそれぞれのページを完全に削除できるボタンを押すと完全に削除できて、そのページの編集ボタンを押すとページ一覧に戻すこともできる。

ゴミ箱にファイルを移動する際は、ページが保存されている kiji というフォルダから trash というフォルダにファイルが移動し、ゴミ箱に入っているページ一覧に移動したページが表示されるようになる。

(5) 管理者メニュー

このソフトでできる機能は、管理者モードとユーザーモードで違っている。ログインしていない状態では、それぞれのページの閲覧とページが一覧で表示されるページの閲覧のみが可能になっていて、それ以外の機能はメニューに表示されていない。

トップページの左側メニューの下側にログインページへのリンクがあり、それをクリックすると、管理者モードにログインできるページ (図 4 参照) に移動する。こちらで設定した、「Okame1nko」というパス

ワードをテキストボックスに書き込み、ログインボタンを押すと、管理者モードに切り替わる。

(6) セキュリティについて

(i) Cookie の設定

管理者モードにログインする機能には、Cookie が設定されている。Cookie とは、Web サイトの中で、Web ブラウザを通してサイト利用者のコンピュータに一時的にデータを書き込んで保存させる仕組みのことであり、Web サーバ側が指定したデータの保存が可能な機能である。

一つの Cookie には 4096 バイトのデータを記憶しておくことができ、ひとつのサーバーでひとつのコンピュータに 20 個の Cookie を発行することが可能である。作成できる Cookie の総数は 300 個までであり、それ以上になると古いものから順に削除されていく。また、Cookie には有効期限が設定されていて、それを超えてしまうと破棄されてしまう。

Cookie がよく使われていることの例としては、利用者の識別をしたり、特定のページへの訪問回数などを記憶しておいてアクセスカウンターなどにその回数を表示させるなどといったことである。

今回、この Cookie を設定している機能としては、正しいパスワードを入力してログインしたという情報を、ユーザーのコンピュータに記憶させておくということである。管理者モードにログインして、ログアウトしていない状態で一旦別のページに移動してからもう一度このソフトに戻ってくると、管理者モードにログインしたままの状態になっている。

この機能を設定するには、

```
<?php
define("PASSWORD", "Okame1nko");
if (!(isset($_COOKIE["TEST"]) && $_COOKIE["TEST"] !=
  null && md5(PASSWORD) === $_COOKIE["TEST"])){
  header("Location:wiki.php?fn=トップページ.txt");
}
```

と書き込まれている「cookie.php」というファイルを作成しておき、管理者モードのメインページである「wiki_admin.php」というファイルの最初に、「cookie.php」を require で呼び出すことで、ログイン状態を記憶しておく Cookie を設定することができる。

また、管理者のみが利用できる新規ページ作成機能や、ページ編集機能、ページ編集後のプレビュー機能、ページ削除機能、メニュー編集のプログラムの一番初めの行でも「cookie.php」を require で呼び出しているため、それぞれのページで Cookie を用いてログインされているかの確認をしている。

(ii) ログアウト機能

ログアウト処理については、管理者モードの左側サイドメニュー内に form を使ってログアウトするボタンを表示させ、それを押すとログアウトができる仕組みになっている。ソースコードとしては、

```
<form action="logout.php" method="post">
<p><button type="submit" name="logout">ログアウト</button></p>
</form>
```

と書き、サイドメニューの下にログアウト用ボタンを設置する。そして、「logout.php」の中には、

```
<?php
if(isset($_POST['logout'])){
    setcookie("TEST", "", time() - 3600);
    header("Location:wiki.php?fn=トップページ.txt");
}
?>
```

と書き、Cookie を消すことでログアウト処理が完了となる。

(iii) URL について

今回のソフトでは、URL の最後にファイル名が指定されていない場合にも、ユーザーモードのトップページが表示されるよう工夫をしている。「index.html」というファイルをつくり、その中に

```
<html>
<head>
<meta http-equiv="refresh" content="0; url=wiki.php?fn=
トップページ.txt>
</head>
</html>
```

と書くことで、一瞬でトップページに移動させることができる。これは、インターネット上でもよく目にする、「○秒後に自動的にページが移動します」といったものにも使われており、何秒でページ移動するかを指定することもできる。

(7) require の設定

今回、header や footer、左側のメニューや、右側の最新ページ一覧は、require という関数を使って呼び出している。すべてのファイルに逐一 header や footer を書き込んでいるととてもごちゃごちゃとしてしまい、見づらくなってしまう。

そこで、require を使えば、指定されたファイルを呼び出すことができる。div タグで囲めば、その範囲だけを CSS でデザインを個別で設定することも可能になる。具体的には、ヘッダーを呼び出す場合

```
<div id="header">
<?php
require "header_user.php";
?>
</div>
```

このように書くと、header_user.php 内に書かれていることを呼び出すことができる。そして、CSS 内に #header と書けば、header のみのデザインの変更をすることができる。今回 require で呼び出しているファイルは、前述の管理者画面の場合、

- header.php (ヘッダー)
- menu.php (左側メニュー)
- ichiran.php (右側最新ページ一覧)
- wikikihou.php (Wiki 構文)
- footer.php (フッター)
- cookie.php (ログインの際の Cookie)

の 5 つである。header と左側メニューと右側の最新ページ一覧につい

では、ログイン前のユーザー画面ではそれぞれ、「header_user.php」、「menu_user.php」、「ichiran_user.php」となっている。

(8) Wiki 構文について

Wiki 構文の変換は、前述の require で wikikihou.php という変換用のファイルを読み込んでいる。

このファイルを作る過程としては、私が様々な Wiki ソフトの Wiki 構文を参考にしながら、それをインライン要素とブロック要素に分けて Wiki 構文の案を考えた。どのように書けばどのように変換されるか、という Wiki 構文の案を先生に提出し、依頼して変換するファイルを作っていた。ファイル作成の中で、より良くなるように私の案を先生が少し変更された。

そして一度、仮にマークダウン記法で変換できるファイルを require で読み込み、試しに動作確認をした。正しく動作することを確認してから、require で読み込むものを今回先生に作成していただいた Wiki 構文のファイルに変えることで、Wiki 構文が使用できるようになった。

(9) デザインの設定

このソフトのデザインは、いろいろな Wiki ソフトやサービスの画面を見て一番一般的だと思われるデザインにしている。左側にメニューとログイン画面へのリンクを配置し、右側に最新ページ一覧を配置した(図1参照)。

右側の最新ページは最新のもの 10 個までが表示されており、それ以下になると少し画面の上下が縮まるようになっている。ページが 10 個以上になると、古いものから表示されなくなり、ページ一覧もしくは

「すべてのページを表示する」というリンクから確認することができる。
また、左側メニューはどんな長さにも対応できるように、padding と margin の bottom で調節している。

3 アンケート

(1) アンケートの結果

今回、この Wiki ソフトをゼミの三回生に実際に使用してもらい、使ってみた感想や意見、不満などについてアンケートをとった。アンケートの項目は以下とした。

1. このソフトは使いやすいと感じたか
2. このソフトについて
 - ・ 良かった点
 - ・ 悪かった点
3. このソフトのデザインについて
 - ・ 良かった点
 - ・ 悪かった点
4. このソフトに付け加えるべき点・改善点など

回答者は 8 名で、1 の質問は「はい」「いいえ」の二択で回答してもらった。8 名中 5 名が「はい」と答え、2 名が「いいえ」と答え、1 名がどちらも選んでいなかった。2 の質問では、良かった点も悪かった点も両方書いてほしいとアンケートに記載しておいたので、多くの意見をもらうことができた。

(i) 良かった点

このソフトの良かった点としては、

- ・ ページの削除ボタンを押した際、ページがゴミ箱に一度入って、ゴ

ミ箱に入っているページ一覧からもう一度ページ一覧に戻せるといった機能がある点

- Wiki 構文が使えることで、HTML やプログラミングの知識が全くなくても、文字の大きさや色が変わえられて、簡単な表や箱囲いなどができる点
- デザインがシンプルでスッキリしていて見やすい点
- 色合いが落ち着いていて目が疲れない点

などの意見が挙げられており、その中で一番多かったのがページ削除の機能についてであった。ワンクッションあるので間違えて記事を消してしまうことがないことや、このページは必要ないと判断してゴミ箱に入れてしまっても、後で必要になったときにまた記事を元あったところへ戻せるところがよいと評価された。

(ii) 悪かった点

対して、このソフトの悪かった点としては、

- メニュー編集でページ一覧や新規ページ作成などといった項目も編集できてしまい、もし誤って削除してしまうと Wiki として利用できなくなってしまうという点
- うまく変換されない Wiki 構文があるという点
- メインの部分がすべて文字が左寄せに設定されていたので見づらかったという点
- ページを作成・編集、メニューの編集などをする際のテキストボックスの中の文字が小さく、編集しづらいという点
- ページ一覧には、それぞれのページの名前と更新日が書かれているだけなので、記事が多くなっていくと見づらくなってしまいそうだという点

以上の点が挙げられた。うまく変換されない Wiki 構文がいくつかあることについての意見は非常に多かった。これについては後述する。

(iii) 付け加えるべき機能

3つ目に、このソフトに付け加えるべき機能について意見を出してもらった。

- ・ 新規ページを投稿する際に表示されるプレビューは、Wiki 構文が記入したままそのまま表示されるのではなく、ページに表示される状態で見られるようにするほうがよい
- ・ 使用できる Wiki 構文の一覧のページがあったほうがよい
- ・ CSS の編集をソフトの中でできるようにしてほしい
- ・ ページ作成・編集の際のテキストボックスの上に、押すだけで Wiki 構文が挿入されるボタンがほしい

このソフトを使ってもらった際には、使用できる Wiki 構文のリストを紙に印刷してアンケートと一緒に配布した。しかし、このソフトの中に Wiki 構文のページを作ることによって、コピーして利用できるので構文のタイプミスも減り、使いやすいとのことであった。

(2) アンケートの評価をふまえて

アンケートの中の「Wiki 構文がうまく変換されない」との意見が多かったのは、私が先生に依頼した Wiki 構文の記法の案が間違っていたことが1つの要因として考えられる。

* (アスタリスク) を用いた構文が重複しており、箇条書きと文字を斜体にする際の両方に使われていた。行の頭に*を書くと、その行が箇条書きになるのだが、文字を斜体にして強調させたい際も*で対象の文字を挟むため、行頭で文字を強調させたい場合に、箇条書きになってし

まうといった問題が起きていた。また、文字を太字にして強調させたい場合も、** (アスタリスク 2 つ) で対象の文字を挟むため、文字を斜体に装飾する際と同様の問題が起きていた。

そして、Wiki 構文がうまく変換されなかったもう一つの要因は、Wiki 構文を使用した行の上下は開けておかなければならないということ、実際に使ってもらう際に注意点としてきちんと伝えなかったことであった。それが原因で、改行せずに続けて異なる Wiki 構文をいくつも試してしまい、きちんと表示されないなどを行った問題が起きてしまった。

また、CSS の編集については、CSS ファイル (Wiki.css) を直接書き換えればデザインの変更も可能だが、意見をいくつかもらって、確かにソフトの機能の一つとして、テキストボックスで編集できるようにしたほうが万人にわかりやすいソフトになると気づいた。

デザイン的な面で悪かった点として挙げられた、メインの文字が左寄せで見づらいという点やテキストボックスの中の文字が小さいといった点は、CSS の中で文字の大きさを変更したり、文字中央寄せや右寄せに編集すれば改善できると思われる。

その他にも、左側サイドメニューをすべて編集できるようにしてしまうと、誤って主要な機能のリンクを消してしまうとうまく Wiki として動かなくなってしまう、といった意見もいくつか挙げられていた。これは、新規ページ作成、ページ一覧などという主要な機能のリンクは左側サイドメニューに固定し、自分で作成したメニューだけを自分で編集できる仕様にすれば解決すると思われる。

または、左側サイドメニューは管理者が作成するメニューのみにして、ヘッダーの上に Wiki で利用できる主要な機能の、変更不可能なメニューを新しく作るといった対策をとれば、メニューを編集しても

Wiki の機能が使えなくなってしまうこともない。

4 自己評価

(1) 良かった点

この Wiki ソフトを作成して、自分の中で良かったと思えるところは、セキュリティ面で Cookie を用いることができたことである。Cookie を設定したために、管理者モードのページの初めにログインしているかの確認ができるようになっている。そのおかげで、ログインしていない状態で管理者しか利用できないページの URL をアドレスバーに直接書き込んでも、そのページは開かずにユーザーページのトップページが開かれる。

また、いくつかのページを閲覧している間に、リンクなどから別のサイトにいってしまっても、ログインしたという Cookie の情報がコンピュータに残っていればログインし直す必要はない。ここが、今回のソフト制作の中で一番良かった点だと言える。

また、ゴミ箱というフォルダを作ったことで、誤ってファイルを削除してしまうリスクが減ったことも良かった点といえる。ページを削除するには、まず自分の作ったページをゴミ箱に移動させる際に、削除ボタンを押す。そして出てきたポップアップでゴミ箱に移動させてよいかと確認をし、OK ボタンを押す。そのあと、ゴミ箱へ移動し完全に削除するボタンを押すといった 3 ステップを踏むことになる。そのため、操作ミスでファイルを削除してしまうことはほとんどないといえる。

(2) 悪かった点

このソフト上で、悪かったと思えるところは、まず、ページタイトルの編集の際、必ず最後に「.txt」をつけなくてはならないことと、「.txt」をつけてページタイトルを変更しても、変更前のページはそのまま残ってしまい、変更後のタイトルのページが新しく出来ているだけといったことである。

他にも、管理者モードとユーザーモードで分けているにもかかわらず、コンピュータに保存されているファイルを直接書き換えることはだれでも出来てしまうといった点も、悪かった点と言える。

時間と知識があれば今回の制作物に付け加えたかった機能はいくつかあり、まず一つ目は、このソフト上でデザインを変更できるような機能である。この機能があれば、CSS を編集できる知識がない方でも、Wiki のデザインを自由に簡単に変えられることができる。

二つ目はページ一覧での並び替え機能である。今の状態では、古いページから順に新しいものへ下から上に表示されているが、新しいページから順番に並び替えたり、五十音順に並び替えたりという項目をセレクトボックスで選択することで、簡単に並び替えができるようになるとできるともっと便利で見やすいページ一覧になるのではないかと考える。また、ページの検索機能も追加できれば、もっと使いやすいソフトになると考えられる。

5 おわりに

今後の課題としては、ソフトについて何もわからない状態から使う側の気持ちになって制作することが挙げられる。今回、自分だけが理解している Wiki 構文の注意点などを言い忘れたり、Wiki 構文のコピーが

——安全で簡単な Wiki ソフトの作成について——

できるページを作らなかつたりと、使う側から見ると不親切な作りのソフトになっているという失敗をたくさんしている。

また、アンケートなどをとったあと、意見を整理している際に、たくさん「この機能も付け加えたかった」という案が頭に浮かんできた。ソフトの制作をしている最中は、自分の知識でできる範囲のことしか考えられていなかったが、一旦制作から離れてみて思いつくことも多かった。知らない間に、自分のできることに絞ってソフトを制作してしまっていたが、これでは、人の役に立つものを作っているとは言えないのではないだろうか。なんとか使用できてはいるが、不完全な部分もまだいくつも残っている。

今後は、これまで以上に貪欲に、便利な機能を追加するための情報収集をし、ソフトを使う側からみても使いやすいソフトにし、実用化してゼミの Wiki として利用できるレベルまで使いやすくなるように試行錯誤していきたいと考えている。

注

- (1) ブログを作成するアプリケーションのひとつで、日本だけでなく世界中で広く利用されている。
- (2) <http://atwiki.jp>
- (3) <http://wiki.fc2.com>
- (4) <http://pukiwiki.sourceforge.jp/?FrontPage>
- (5) <http://wiki.seesaa.jp>
- (6) <http://www.mediawiki.org/wiki/MediaWiki/ja>
- (7) 世界中で最もよく利用されている、オープンソースのデータベースのうちの一つ。高速で、非商用利用であれば無償で利用できる。

文献表

IT 用語辞典

<http://e-words.jp>