Flashy Site Builder

By

Christina J Robinson

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Engineering Technology

University of Cincinnati
College of Applied Science

March 2005

Flashy Site Builder

by

Christina J Robinson

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Engineering Technology

© Copyright 2002 Christina Robinson

The contents of this document are under copyright of the author. It may not be reproduced and distributed in whole or in part without the written permission of the author.

C Robinson	3-11-05
Christina J Robinson	March 11, 2005
Thomas E Wulf	3-11-05
Professor Tom Wulf, Faculty Advisor	March 11, 2005
James F Sullivan	11 March 2005
James Sullivan, Department Head	March 11, 2005

Flashy Site Builder Abstract 1
1. Problem Description and Intended Use 2
1.1 The Need for Flash 2
1.2 The Need for a Solution 2
1.3 Flashy Site Builder 2
2. Flashy Site Builder User Profile 2
2.1 Designated Audience 2
2.2 Flashy Site Builder Users 3
2.3 Help Menu 6
2.4 User Requirements 6
3. Flashy Site Builder Design 7
3.1 Flashy Site Builder Graphical User Interface Design 7
3.2 Color Scheme 7
3.3 Flashy Site Builder Navigation 7
3.4 Buttons and Graphical Symbols 10
4. Flashy Site Builder Application Design 10
4.1 Technologies Used 10
4.2 Reasons for Chosen Technologies 11
5. Create Flashy Site Builder Presentations 11
6. Deliverables 13
6.1 Flashy Site Builder Templates 13
6.2 Flashy Site Builder Images 13
6.3 Flashy Site Builder Sound 13
6.4 Flashy Site Builder Text 13
6.5 Publish and Modify Flash Presentations 13
7. Test Plan 14
7.1 Testing Audience 14
7.2 Procedures for Testing 14
8. Flashy Site Builder Budget 14
8.1 Total Cost 14

9. FSB Timeline 15
9.1 Senior Design II 16
9.2 Senior Design III 17
10 Code Snippet 17
Resources 25

Flashy Site Builder Abstract

Flashy Site Builder (FSB) is an online flash website creator and site management system. FSB is created to generate astonishing, high end multimedia flash websites and stand-alone multimedia presentations. The flash presentations work on any computer anytime anywhere in the world with nothing to install, upgrade, or buy except the free flash plug-in. Using flash, developers can create rich interactive experiences to achieve superior results. But who has time to learn Macromedia Flash? Novices need the ability to create rich content and applications across desktop devices. Not everyone has time or the programming skills required to learn how to develop flash websites or presentations. Users need an extremely easy way to build flash websites and presentations. Users need a flash website builder.

This paper discusses how FSB is an online application that combines multimedia, database, and Web technologies to allow users to dynamically design flash websites and presentation. The paper discusses the user groups, FSB navigation and application design, along with the project deliverables, testing and timeline.

Flashy Site Builder

1. Problem Description and Intended Use

1.1 The Need for Flash

Standard websites can only be viewed in a Web browser, but Macromedia Flash allows the ability to create rich content and applications across desktops and devices. Blue Pacific's website reads several studies have shown rich media produces more engaging and compelling user experiences than static content like HTML. Using flash, a developer can create rich interactive experiences to achieve superior results. Flash is a full-featured animation technology that can run with or without a Web browser. According to 10 Minute Flash, thousands of high-end sites depend on flash for viewing content on the Internet, causing Macromedia Flash to be the world's most pervasive rich client.

1.2 The Need for a Solution

Novices need the ability to create rich content and application across desktop devices. Not everyone has time or the programming skills required to learn how to develop flash websites or presentations. Users need an extremely easy way to build flash websites and presentations. Users need a flash website builder.

1.3 Flashy Site Builder

Flashy Site Builder (FSB) is an online flash website creator and site management system. FSB is created to generate astonishing, high end multimedia flash websites and stand-alone multimedia presentations. FSB is an online application that combines multimedia, database, and Web technologies to allow users to dynamically design flash websites and presentation. Using a flash enabled Web browser, FSB works on any computer anytime anywhere in the world with nothing to install, upgrade, or buy.

2. Flashy Site Builder User Profile

2.1 Designated Audiences

FSB is easy to use. The online multimedia application allows anyone, even a person who knows nothing about programming or design, to create a professional looking Macromedia Flash website or presentation in minutes. With FSB even a novice can design a professional

multimedia website or presentation in minutes. FSB is as simple as a point and click. The designated audiences for FSB are users who want to create:

- personal/professional websites
- professional resume presentations
- photo album presentations

2.2 Flashy Site Builder Users

Any persons with the ability to use a mouse and keyboard can use FSB. FSB has two user groups; new and experienced users. FSB starts by playing the welcome screen. Figure 1 shows the welcome screen.

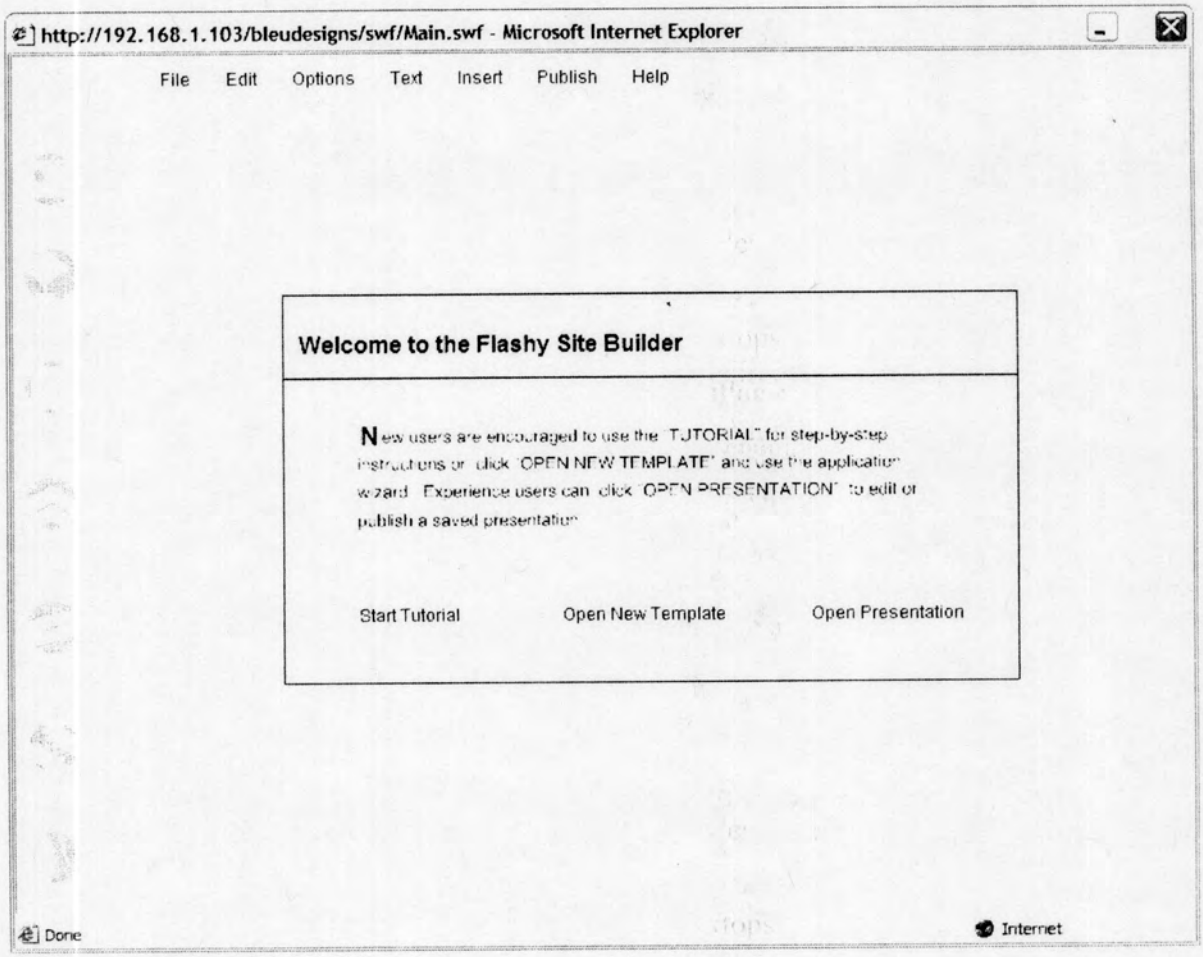

Figure 1: Welcome Screen

Figure 1 also illustrates the following three buttons:

- Start Tutorial: Users unfamiliar with FSB or new users are encouraged to click the "*Start Tutorial*" button. The easy to follow tutorial provides step-by-step instructions to create a flashy presentation. The tutorial guides new users through the design process by suggesting what options and text to use in order to become familiar with FSB. Users can click and drag or close the tutorial window whenever necessary.
- Open New Template: The "*Open New Template*" button is suggested to be utilized by new users or users who have already completed the tutorial, and are familiar with FSB. When a new user chooses to start a new template they can use the HELP drop down menu to start the tutorial whenever necessary.
- Open Existing Presentation: Only users who have a saved presentation or an account created with FSB are able to successfully click the "*Open File*" button. This button allows users to load their exiting presentation into FSB.

After a user clicks one of the three buttons on the welcome screen the user can be classified as either a new user or experienced user. Chart 1 exemplifies the break down of the two user groups: new and experienced. The FSB user group chart illustrates how new users who are unfamiliar with FSB start by using the tutorial. And how experienced users are individuals familiar with the tutorial, but are either returning to edit or publish a saved project, or opted to close the tutorial.

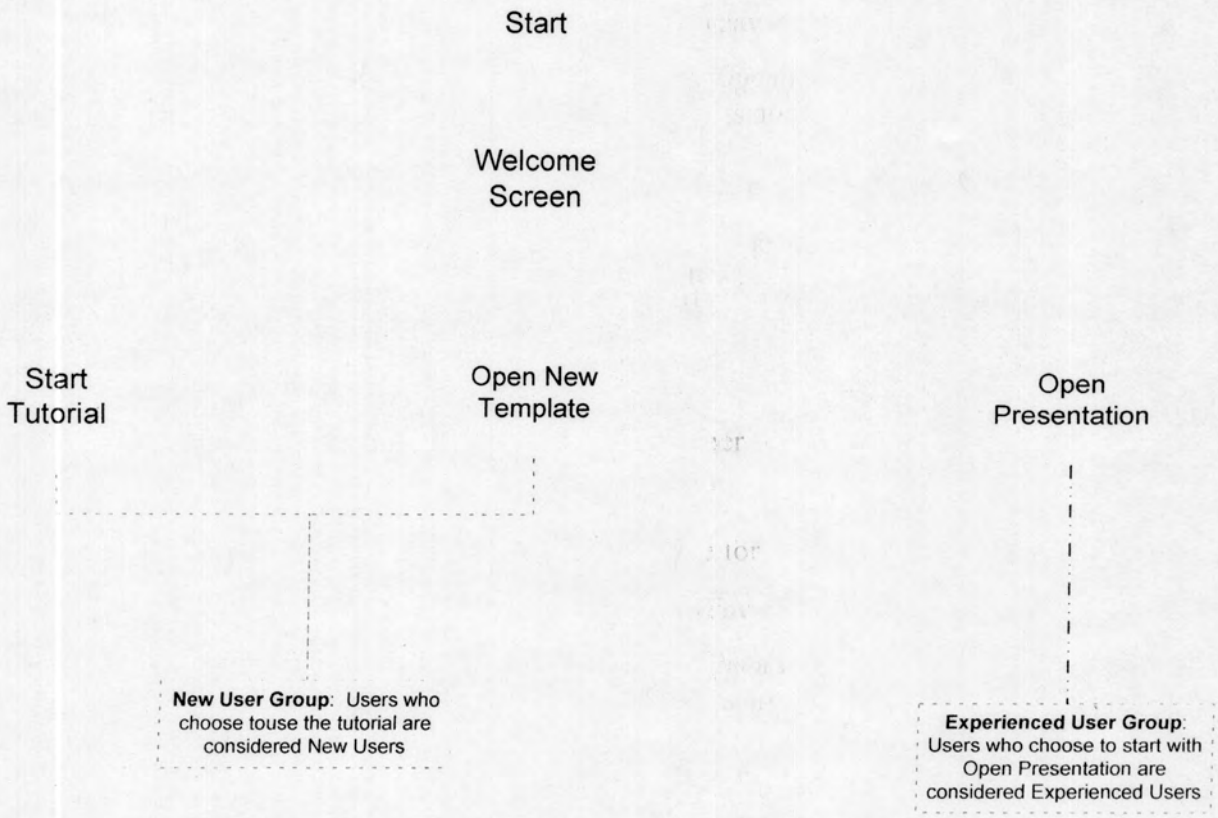

Chart 1: FSB User Groups

FSB allows users to change user groups at any point during the creation of a presentation. A new user can close the tutorial and instantly become an experienced user. Vice versa, experience users can use the help menu to start the tutorial, changing the experienced user to a new user. Figure 2 illustrates the help menu and the options available for both user groups. An experienced user can use help to start the tutorial for FSB.

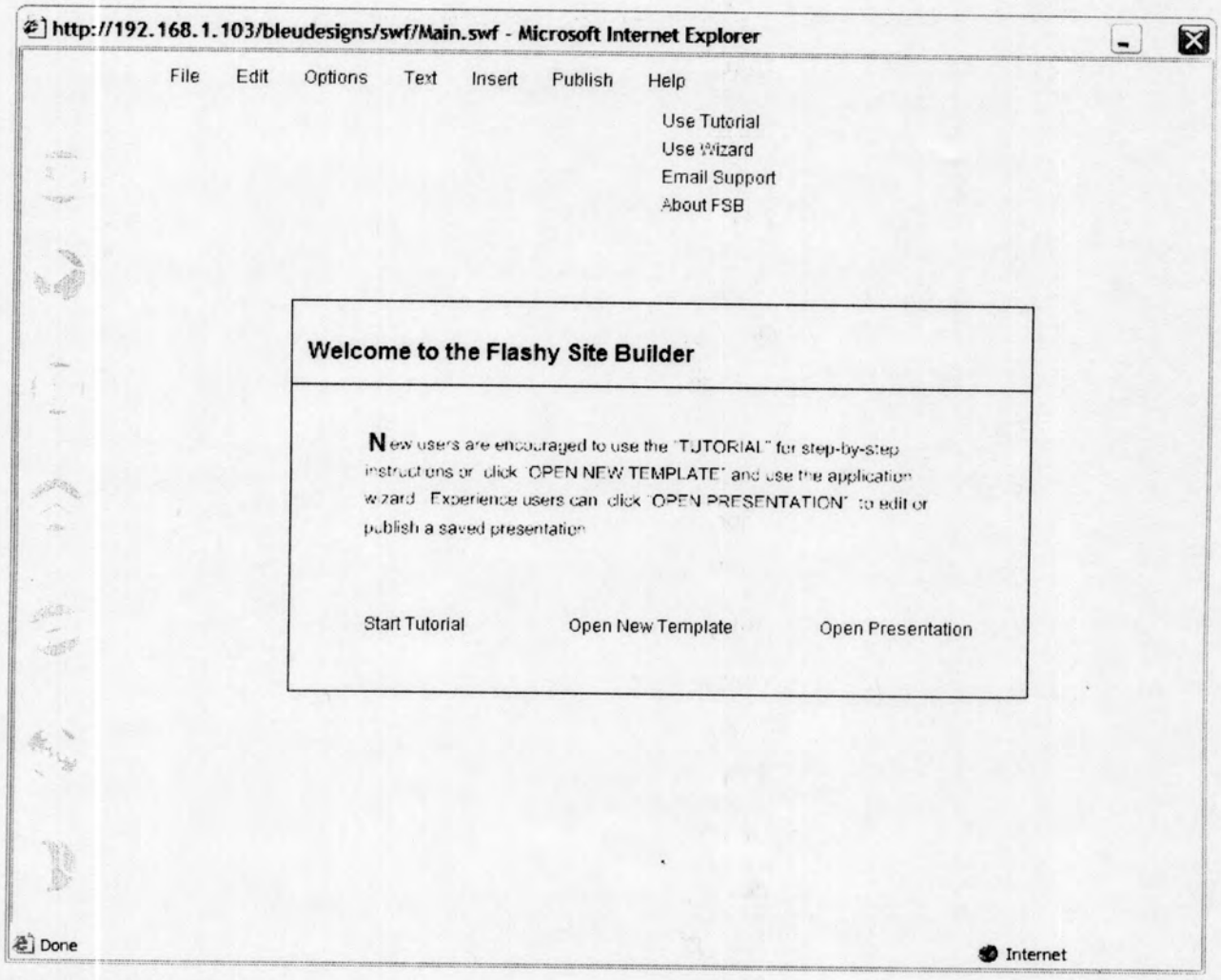

Figure 2: Help Menu

2.3 Help Menu

The help menu is also how users get the necessary information about FSB. By using the help menu users are not only able to access the tutorial, but users can also send an email to FSB support.

2.4 User Requirements

Before executing FSB, two requirements must be achieved. The user must be using Microsoft Internet Explorer or Netscape Navigator as their Web browser. And the Web browser must have the latest Macromedia Flash plug-in installed. Currently Macromedia Flash Player 7.0 is the latest plug-in. According to http://www.macromedia.com 97% of all Internet-enabled desktops and many popular devices already have the free plug-in, either because it was pre-installed on the

browser or the user downloaded it. Some users may need to upgrade their plug-in to Flash Player 7.0 before using FSB. There is no other software to buy or install.

3. Flashy Site Builder Design

3.1 Flashy Site Builder Graphical User Interface Design

FSB is built 100% with flash, and uses simple drop down menus. Building upon my design and development skills FSB is built to be the next generation of Web applications. The entire FSB application uses multimedia, database, and Web technologies. FSBs' conceptual design is based off the components available in the multimedia webpage builder from 10 Minute Flash. The design of FSB makes it possible to construct a multimedia flash presentation with added features like sound, images, and color changing. FSB makes using those added features an effortless experienced for both new and experienced FSB users.

3.2 Color Scheme

The FSB application is designed using the following classic colors to create a subtle color scheme:

- Grey: color of the drop down menu
- White: background color
- Blue: color of the buttons

3.3 Flashy Site Builder Navigation

Users launch FSB from a webpage, Figure 3 shows the webpage.

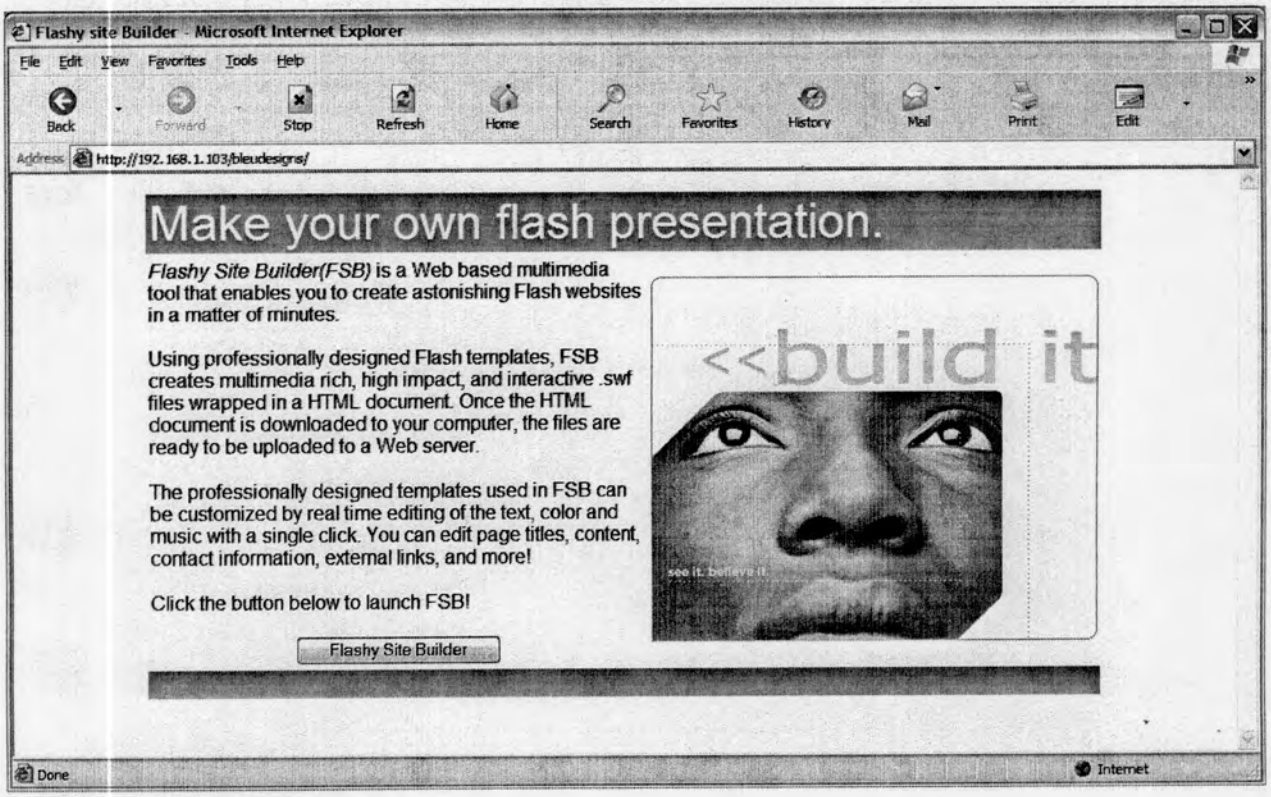

Figure 3: Webpage to Launch FSB

Users click the "Flashy Site Builder" button, and FSB launches in a new window presenting the user with the FSB welcome screen. FSB adopted the idea of using a dropdown menu and buttons for experienced users to navigate through FSB. New users are guided through the exploration of the dropdown menu and buttons through the tutorial and wizard. Figure 4 shows FSB being used by an experienced user.

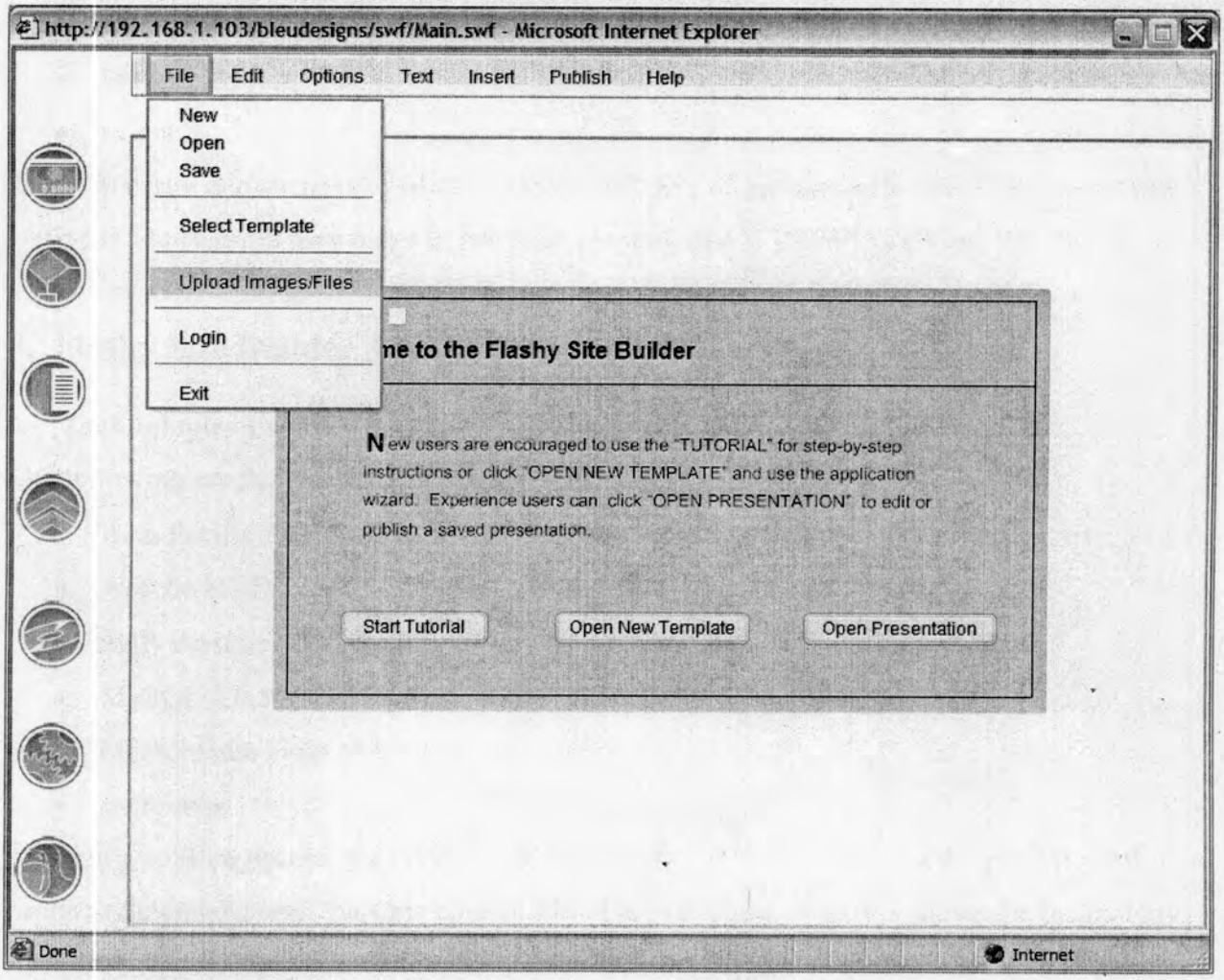

Figure 4: Dropdown Menu and Navigation Buttons

The dropdown menu bar shows options for the following:

- File
- Edit
- Help

3.4 Buttons and Graphical Symbols

The buttons on the left are shortcut commands to options in the dropdown menu. The buttons represent key points for frequent user interaction. Many of the button commands are listed under the File dropdown menu option. The following are the button options:

- Select Template
- Start a New Project

- Open a Saved Project
- Publish Project
- Login

The FSB menu allows users to edit all aspects including direct manipulations of the templates. Users see the changes they make in real time, creating true WYSIWYG (What-You-See-Is-What-You-Get) editing.

4. Flashy Site Builder Application Design

4.1 Technologies Used

The following are the technologies used to create FSB:

- Blue Pacific PHP Turbine 7 for Windows (demo version)
- Apache HTTPD Server Version 2.0.47
- PHP Version 4.3.3
- MySQL 4.0.15
- Macromedia Flash MX Professional 2004
- JavaScript

According to Blue Pacific, the creator of PHP Turbine 7, PHP Turbine 7 is the perfect solution to generate rich media with dynamic content from PHP scripting. Figure 5 shows the technology flow process and demonstrates how the technologies work together to develop an astonishing, high end multimedia flash presentation.

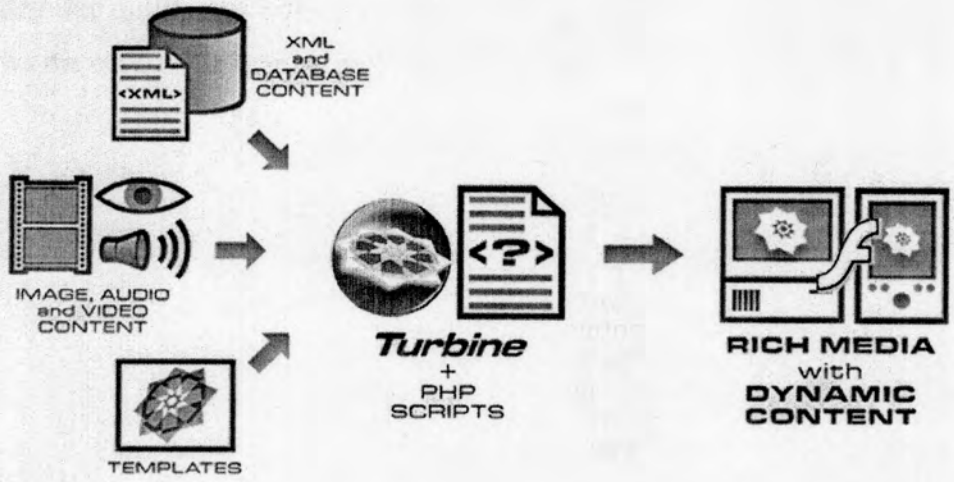

Figure 5: Technology Flow Process

The technology flow process shows how Turbine uses Macromedia Flash templates to integrate dynamic content from a variety of images, audio formats, and MySQL to create a rich media flash file with dynamic content.

4.2 Reasons for Chosen Technologies

Apache, MySQL and PHP are used because each technology is opened source software, and work well together. The MySQL website states, MySQL has become the most popular open source and fastest growing database in the industry, I selected MySQL because of; reliability, performance, and ease of use and deployment. I chose to use Blue Pacific PHP Turbine 7 because Turbine:

- Generates rich formats in flash
- Has a short learning curve due to PHP integration
- Uses flash files as templates
- Can use flexible XML templates for powerful automation
- Allows integration from MySQL

5. Create Flashy Site Builder Presentations

Once FSB is launched the user has access to a library of professionally designed templates and components. The user has full control of when and how animations should play. Because of the easy to use interface, users point and click their way to a beautiful flash website in a matter of minutes. By directly manipulating the templates in a WYSIWYG interface, users are able to customize their site millions of different ways to create a one-in-a-million dynamic presentation. Chart 2 shows the complete organizational scheme to creating a flash presentation.

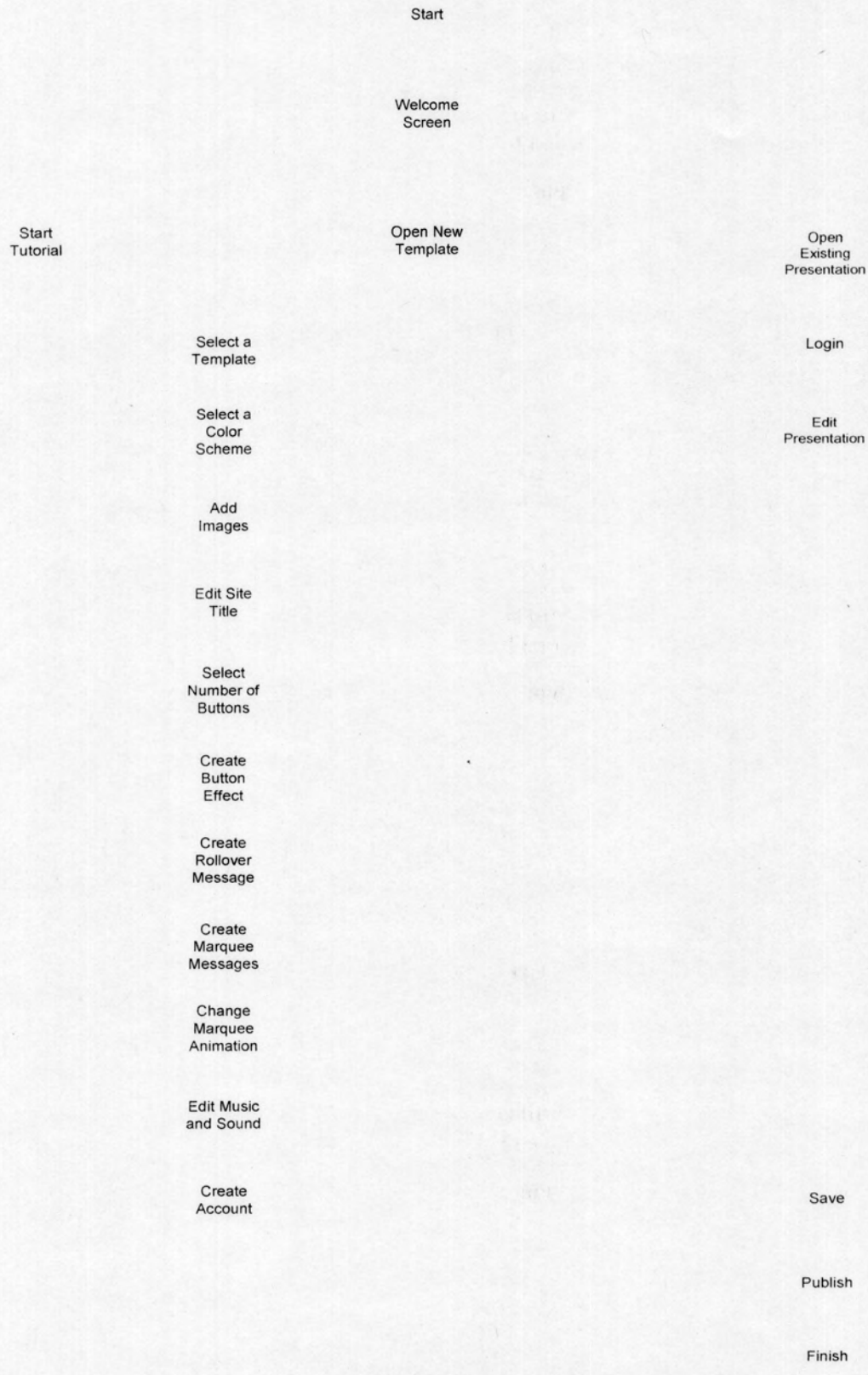

Chart 2: Organizational scheme Creating a Flash Presentation

6. Deliverables

FSB will deliver users the ability to

- Design professional templates
- Use stock images
- Dynamically edit text
- Publish a website
- Login information to a database

6.1 Flashy Site Builder Template

- FBS takes care of all the hard work. The template come standard with dynamic motion effects to captivate any audience.

FSB allow direct manipulation to all templates by allowing users to add and modify content information, and the ability to change between different templates without losing any information.

6.2 Flashy Site Builder Images

FSB offers professional stock images. The image includes seven categories:

Animals

Business

Cultural

Fun

People

Places

Travel

6.4 Flashy Site Builder Text

FSB uses dynamic text which features word processor-like rich text editing. The text color size, font, and alignment can be changed.

6.5 Publish and Modify Flash Presentations

Users can save presentations and modify published presentations even after created. After creating a one-in-a-million flashy site or presentation users can download the .swf and host it virtually anywhere or burn to a CD-ROM. If the user needs to make a change, all they need to do

is revisit the FSB application, login, and make the change; the site is updated instantly. One click will generate a dynamically created flashy presentation for the entire world to view.

7. Test Plan

7.1 Testing Audience

The procedure for testing FSB allowed a small audience to test the application. The test audience include at five members, and the audience was comprised of:

- Professionals with moderated computer skills
- Information Technologist
- Users with little computer literacy

7.2 Procedures for Testing

Each member of the audience test group was asked to create a presentation using FSB. First the test subject was asked to create a presentation as a new user, using the tutorial. Then the test subject created a website as an experienced user. After each presentation was completed the test subject was required to complete an oral evaluation. The evaluation consisted of the following questions:

- Was the tutorial useful?
- Are there any application options that should be added / deleted?
- Could FSB be recommended to a friend?
- Rate the overall usability of FSB (application color scheme, button choices, and menu options).

After the first stage of testing was completed, modifications were made based on audience analysis. The second stage of testing consisted of the same test group, with the same requirements, and a final edit on the application was made.

8. FSB Budget

8.1 Total Cost

The total cost for developing FSB is $699. Table 1 illustrates the break down of the cost for the specific software needs.

Table 1: Budget

Item	Cost
Apache Server *	$0.00
PHP *	$0.00
MySQL *	$0.00
Blue Pacific PHP Turbine 7 for Windows **	$0.00
Macromedia Flash MX Professional 2004	$699.00
Total	$699.00

* Open source software
** Demo version

9. FSB Timeline

The schedule for Senior Design II and III is listed below. The schedule shows the completed project milestones. I devoted approx 20 hours per week to the development of the FSB.

9.1 Senior Design II

Monday, October 11:

- Approval was granted to modify the original project proposed in Senior Design I
- Review Draft of Project Description and Intended Use and User Profile with Advisor

Wednesday, October 13:

- Assignment Due: Draft of Project Description and Intended Use and User Profile – Signed by both advisor and learner
    - Project Milestones
        - Configure Apache Server
        - Configure PHP
        - Configure MySQL

Monday, October 18:

- Review Progress Report: 1 with Advisor

Wednesday, October 20:

- Independent Consultations with Advisor
- Assignment Due: Progress Report 1 – Signed by both advisor and learner
- Assignment Returned: Draft of Project Description and Intended Use and User

- **Project Milestones**
  - Research Software (Blue Pacific, PHP, and MySQL) to Create .swf Files

Wednesday, October 27:

- Independent Consultations with Advisor
- Review Draft of Design Freeze

Wednesday, November 3:

- Assignment Due: Draft of Design Freeze – Signed by both advisor and learner

Wednesday, November 10:

- Review Progress Report: 2 with Advisor

Wednesday, November 10:

- Assignment Returned: Draft of Design Freeze
- Discussion of Final Presentation Requirements
  - **Project Milestones**
    - Complete application GUI layout.

Monday, November 15:

- **Project Milestones**
  - Connect flash, MySQL, and PHP to work together, and successfully complete the login frames
  - Display an icon in a menu component

Wednesday, November 17:

- Assignment Due: Progress Report 2 – Signed by both advisor and learner

Wednesday, December 1:

- Assignment Due: Design Freeze signed by both the learner and the advisor
  - **Project Milestones**
    - Successfully Pass and Receive PHP Parameters and MySQL values to and from Flash

Monday, December 6:

- **Project Milestones**
  - Converting dynamic object to .swf file on server
  - Add and subtract pages without loosing information
  - Change between different templates without losing any information

- Complete Template
- Successfully edit template using application menu

Wednesday, December 8:

- Oral Presentation of Proof of Concept/Working Quick Prototype to Faculty and Fellow Learners

9.2 Senior Design III

A detailed timeline was proposed once the requirements for Senior Design III were given.

Week 1-6

- **Project Milestones**
  - Successful wrapping .swf file in a XHTML wrapper using PHP
  - Design a professional template
  - Build stock images library
  - Dynamically edit text
  - Create the ability to publish a website or save the presentation to the local pc

Week 7-8

- **Project Milestones**
  - Completed Audience Testing
  - Modified FSB based on Audience Analysis

Week 9-10

- **Project Milestones**
  - Complete second audience testing and final GUI modifications to FSB

March 10, 2005:

- Present Final Project: Oral Presentation of Proof of Concept/Working application to Faculty and Fellow Learners

10.Code Snippet

Flashy Site Builder is the next generation of Web applications. The drop down menu used for FSB makes every function for FSB possible. Without the edit portion of the menu, user would not be able to change the:

- Title

- Button Text
- Page Text

The following are code snippets for the edit portion of the drop down menu:

```
/**********************************
CODE SNIPPET FROM MAIN.SWF
**********************************

/**********************************
Import the required classes. Use the wildcard (*) to import all classes within the mx.controls package.
**********************************/
import mx.controls.*;

//instantiate a new Local Connection object
var menuOption_lc:LocalConnection = new LocalConnection();

// declare variables  instances on the stage
var MainMenu:MenuBar;
var MainWindow:MovieClip;

// declare variables  variables in this script
var file_menu:Menu;
var edit_menu:Menu;
var editListener:Object = new Object();

// create the button text drop down menu
```

```
button_text = edit_menu.addMenuItem({label: "Button Text", instanceName: "button_textInstance"});
button_text.addMenuItem({label:"None", instanceName:"nonebutton_text", type:"radio", selected:true, enabled:true,
groupName:"group4"});
button_text.addMenuItem({label:"Button 1", instanceName:"Button_1", type:"radio", selected:false, enabled:true,
groupName:"group4"});
button_text.addMenuItem({label:"Button 2", instanceName:"Button_2", type:"radio", selected:false, enabled:true,
groupName:"group4"});
button_text.addMenuItem({label:"Button 3", instanceName:"Button_3", type:"radio", selected:false, enabled:true,
groupName:"group4"});
button_text.addMenuItem({label:"Button 4", instanceName:"Button_4", type:"radio", selected:false, enabled:true,
groupName:"group4"});
button_text.addMenuItem({label:"Button 5", instanceName:"Button_5", type:"radio", selected:false, enabled:true,
groupName:"group4"});
button_text.addMenuItem({label:"Button 6", instanceName:"Button_6", type:"radio", selected:false, enabled:true,
groupName:"group4"});
button_text.addMenuItem({label:"Button 7", instanceName:"Button_7", type:"radio", selected:false, enabled:true,
groupName:"group4"});
button_text.addMenuItem({label:"Button 8", instanceName:"Button_8", type:"radio", selected:false, enabled:true,
groupName:"group4"});

/**********************************
```

The listener for EDIT

```
**********************************

editListener.change = function(eventObj:Object):Void{
        switch (eventObj.menuItem) {
        case eventObj.menu.nonebutton_text :
                menuOption_lc.send("conn_SWF","editOptions","None");
                break;
        case eventObj.menu.Button_1:
                menuOption_lc.send("conn_SWF","editOptions","Button 1");
                break;
        case eventObj.menu.Button_2 :
                menuOption_lc.send("conn_SWF","editOptions","Button 2");
                break;
        case eventObj.menu.Button_3 :
                menuOption_lc.send("conn_SWF","editOptions","Button 3");
                break;
        case eventObj.menu.Button_4 :
                menuOption_lc.send("conn_SWF","editOptions","Button 4");
                break;
        case eventObj.menu.Button_5 :
                menuOption_lc.send("conn_SWF","editOptions","Button 5");
                break;
```

```
        }//end switch
};//end function

//register the listeners with the menu
edit_menu.addEventListener("change", editListener);

/**********************************
CODE SNIPPET FROM TEMPLATE.SWF
**********************************/
//instantiate the  instance of the LocalConnection object and make the connection
template_lc = new LocalConnection();
template_lc.connect("conn_SWF");

//declare variables
var _editCmd:String=null;
/*==================
Function for the EDIT menu that will be called from MAIN.SWF
==================*/
template_lc.editOptions = function(_editCmd){
        switch(_editCmd){
        case "none":
                gotoAndStop("start");
```

```
            break;
        case "title":
            gotoAndStop("title");
            break;
        case "None":
            gotoAndStop("start");
            break;
        case "Button 1":
            gotoAndStop("btn1");
            break;
        case "Button 2":
            gotoAndStop("btn2");
            break;
        case "Button 3":
            gotoAndStop("btn3");
            break;
        case "Button 4":
            gotoAndStop("btn4");
            break;
        case "Button 5":
            gotoAndStop("btn5");
            break;
```

```
        }//end switch
};//end function
```

Resources

10 Minute Flash, November 30, 2004.
<http://www.10minflash.com>.

Blue Pacific, November 30, 2004.
<http://www.blue-pacific.com>.

Designer 360, November 30, 2004.
<http://www.designer360.com>.

Macromedia, November 30, 2004.
<http://www.macromedia.com>

MySQL, November 30, 2004.
<http://www.mysql.com>.

Price Watch, November 30, 2004.
<http://www.pricewatch.com>.