E-Commerce:

Designing And Creating An Online Store

Introduction

- About Steve Green Ministries
  - Solo Performance Artist for 19 Years.
  - Released over 26 Records, Several Kids Movies, and Books.
- My History With Steve Green Ministries
  - Started maintaining their existing site 5 years ago

Project Goals

- Goals:
  - Develop a new website to coordinate with the release of Steve's new record, *Somewhere Between.*
  - Create a fully functional online store.
  - Set up a PHP, MySQL, Apache, and OpenSSL testing environment.

Building The New Site

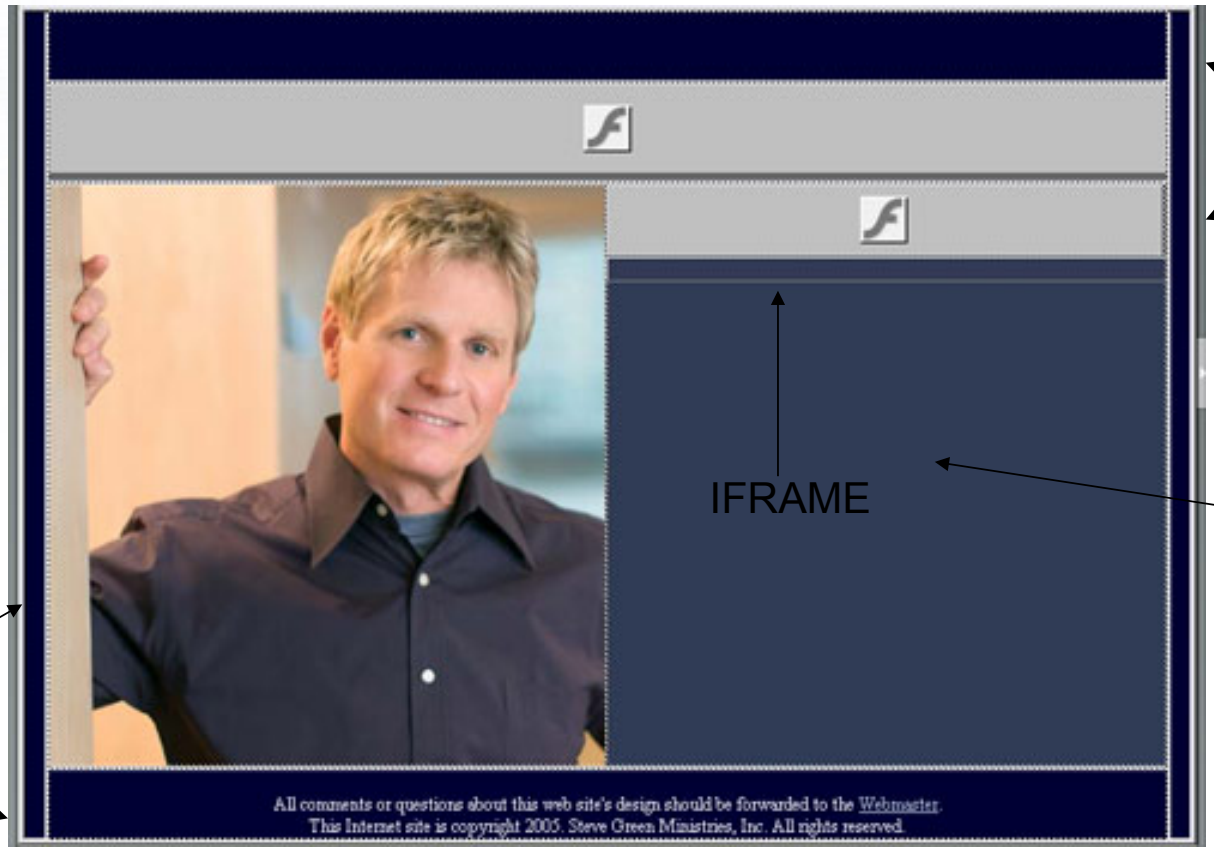
- New Site Requirements:
  - Use frames, so that content could be updated on one part of the page without reloading the entire page
  - Use flash to design and animate menus
  - All coding in HTML – our current server does not support any dynamic content.

Building The New Site

- IFRAMES Example
  - IFRAMES are virtual frames which allow you to position page content anywhere on the screen.
  - IFRAME position is relative to the top and left side of the browser window.
- Problem
  - Because the IFRAME is relative to the window size, it is very difficult to predict its position and design accordingly

Building The New Site

- Solution
  - Create a frameset which would adjust according to the user's browser window.



Building The New Site

- Building Flash Navigation
  - Navigation Menu , Product Menu
- Using Action Script
  - Like Visual Basic in the Microsoft Office environment, Action Script can be used to control almost every aspect of a Flash movie.
  - Products Menu.fla

Building The New Site

- Old Steve Green Ministries Site:
  - Old Site
- New Steve Green Ministries Site:
  - New Site
  - www.stevegreenministries.org

Setting Up A Testing Environment

- LAMP
  - Linux (or Windows)
  - Apache web server
  - MySQL
  - PHP

Setting Up A Testing Environment

- Apache Web Server
  - Automatic Windows Install with basic configuration
  - Modify the configuration file to load the PHP library and map ".php" extension to that library
  - Create virtual hosts (Listen on port 80 and 443)
  - Enable SSL using a dynamically loaded SSL library
  - Configure SSL for port 443

Setting Up A Testing Environment

- PHP
  - Automatic Installation of files with no configuration
  - Modify the PATH to include the PHP library folder
  - Modify php.ini to dynamically load the MCRYPT library (For encryption discussed later)

Setting Up A Testing Environment

- MySQL database server
  - Automatic Windows Installation with minimal configuration
  - Set up user accounts with desired privileges
- Problem
  - PHP lacked the ability to connect to the version of MySQL I installed because the newest version created a new password system.
- Solution
  - Install an older version of MySQL (less secure)

Setting Up A Testing Environment

- SSL – Secure Sockets Layer
  - SSL ensures the security of online transactions between the browser and the server
  - When a browser requests a secured page, the server will send its public key and a certificate of validity.
  - If the certificate is valid, the browser will then use the server's public key to generate a symmetric key.
  - The browser uses the symmetric key to encrypt the information it sends to the server and decrypt the information it receives from the server.

Setting Up A Testing Environment

- SSL Installation
  - OpenSSL – A free, open source alternative
    - Downloaded the source and compiled for Windows in Visual Studio
    - Added the compiled source to the PATH and created an server certificate and a server key.
    - Certificate Authorities such as Verisign charge around $2000.00 for a two year certificate and key.
  - Apache Web Server
    - Modified the configuration file to use the new certificate and key

Building The Store

- Create MySQL Tables
  - Separate different types of information into different tables for security
  - Enforce data integrity with the use of primary and foreign keys
  - Research appropriate data types for each column

Building The Store

- Creating A Products Page
  - Much easier than coding in HTML
  - Created a recordset from an SQL query and populated a table with all the information
  - Stored the file name of the picture in the database for ease of updating
  - Stored the path of the detail page for drill down
  - Categories displayed by a $_GET variable

Products Page Code

Building The Store

- The Shopping Cart
  - 4 Basic Functions of a Shopping Cart:
    - Add an item
    - Update item quantities
    - Delete an item
    - Display cart contents & calculate totals
  - 2 Ways to Store User's Shopping Cart
    - Use PHP Session Variables
    - *Tie to a database

Building The Store

- The Shopping Cart
  - The Database approach to a shopping cart allows for the cart to remain across visits.
  - How to recognize an individual browser:
    - PHP will assign a unique session id to each browser for every visit.
    - To make this session id remain, I stored it into a cookie
  - I simply associated the contents of each cart with the browser's session id.

View Shopping Cart Code

Building The Store

- Adding Users
  - When a customer is ready to checkout, they must either login to an existing account or create an account.
  - To Protect Privacy:
    - Browser caching is turned off. It is possible to gain personal information by reading a browser's cache.
    - All communication between the browser and server is encrypted using SSL.
    - All user data is encrypted before stored in the database.
    - Passwords are irreversibly encrypted using MD5

Building The Store

- Adding Users
  - Every user in put was validated using PHP
  - Add User Code 1 2 3
  - Add User Check Code 1 2 3
- User Authentication
  - Once a user has either signed in or created an account, the user id associated with that account is stored in the browser's cookie.
  - The user id stored in the cookie is encoded and a second variable is created to verify the cookie's integrity.
  - setcookie("client_id", base64_encode($single_row), time() + (3600 * 24))
  - setcookie("chk", base64_encode($single_row.getcid()), time() + (3600 * 24))

Building The Store

- Checkout Process
  - Sign in or create an account
  - Verify cart contents
  - Input Credit Card
  - Display a printable invoice for client records

Building The Store

- MCRYPT
  - Triple DES
    - A revision of DES (Data Encryption Standard)
    - It achieves 168bit encryption by incorporating 3 keys into the encryption process
    - This is a 2 way encryption which utilizes a secret key to encode and decode data

Building The Store

- MCRYPT
  - Problem:
    - MCRYPT returns a random encrypted string. These strings often contain characters which would invalidate an SQL statement. Also, the data types I used in the database would remove trailing spaces on strings and ultimately truncate my encrypted string.
  - Answer:
    - PHP has a function called “addslashes()” which places slashes in front of each problematic character in a string. This makes keeps the encrypted values from causing errors in the SQL statement. Also, I changed the data type of the encrypted columns to “blob” which can store an unlimited amount of binary data.

Building The Store

- The Backend Requirements:
  - View Unfilled Orders
  - View Filled Orders
  - Generate A Printable List Of All Unfilled Orders
  - Delete Orders
  - Clean The Shopping Cart Table
  - View Site Statistics

Building The Store

- The Backend Design
  - The backend must be secure – It has the decryption key
  - Separate Key for User Info and Credit Card Info
  - Store the admin pages on the user's computer – not on the remote server

Building The Store

- Visit The Store
- View Store Code

Overview

- Goals Met:
  - Created a static site with flash navigation and frame linking
  - Set up a testing environment with PHP, MySQL, Apache, and SSL
  - Create an online store, complete with product listing, shopping cart, user accounts, and a backend
  - Ensure information privacy with the use of SSL and data encryption

Sources

- Apache Reference:
  - http://www.apache.org
  - http://www.modssl.org
- PHP Reference:
  - http://www.php.net
- MySQL Reference:
  - http://www.mysql.com
- Shopping Cart Design:
  - http://www.macromedia.com
  - http://www.zend.com/zend/spotlight/php-shopping-cart1.php
  - http://www.phpwebcommerce.com/
  - http://www.phpbuilder.com/columns/evert20000816.php3?page=1
- OpenSSL:
  - http://tud.at/programm/apache-ssl-win32-howto.php3
  - http://www.thompsonbd.com/chris/tutorials/apachessl.html
  - http://www.tldp.org/HOWTO/SSL-Certificates-HOWTO/
- Mcrypt:
  - http://www.phpmag.net/itr/online_artikel/psecom,id,667,nodeid,114.html
  - http://www.edparrish.com/cis165/03f/lesson12.php
  - http://www.linuxjournal.com/node/7237/print
- Triple DES:
  - http://kingkong.me.berkeley.edu/~kenneth/courses/sims250/des.html