Advanced Database Systems Design Document

Client: Artists Workshop
Project: Website & Web-enabled Databases
Code: Html, CSS, Javascript, Lasso, FileMaker Pro
Contact: David Haney
Phone: 480-217-4613
Email: davidhaney@database-systems.com

Background
Artists Workshop has ask ADS to complete a web based ordering system that was designed for their website. The HTML layout is complete, and only needs the lasso code to complete the functionality.

Objectives
To integrate the existing html files provided for the online ordering system with lasso code to make the ordering system functional.

Development Schedule

Development item	*Milestone*
Design Doc Submission for approval	<to be determined>
Database restructuring for website integration	<to be determined>
Catalogue display, search, navigate	<to be determined>
Add to cart, cart display, update, delete	<to be determined>
Order Completion, secure payment integration	<to be determined>
Additional debugging and security testing.	<to be determined>

Development Coordination
Graphic production, design, and HTML have been supplied. ADS is responsible for the modification of databases, routines for maintaining current information in the web databases, assisting with server issues, and modifying the exiting HTML pages to web-enable the database information. ADS will take steps when designing the solution to anticipate the use of a GUI editor for future editing of the provided HTML for the ordering system. The ADS development server will serve as the focal point for the most recent revision of the site. Archives will be made on a regular basis to preserve revisions of the web site.

Site Structure
Provide an overview or outline of entire site with a diagram.

Database Design
FileMaker Pro has been used to create the databases, so they will remain in this format. ADS staff will develop FileMaker Pro databases to contain the information that is to be dynamically displayed on the web. Sample data will be used in these databases during the CGI design process. ADS will retain the most current copy of the web database files until the web site is complete.

Images
All images associated with database records should be stored as web compatible graphics in a folder/directory location on the web server. I would not store the graphics in container fields in the FileMaker Pro database since this slows the return of images from the server and the flexibility of the solution. Also, there can exist a problem when graphics are stored in windows formatted images that may cause a problem when hosted if it is hosted on a Mac. All images must be web-compatible ".jpg" or ".gif" files that are located in a folder labeled "images". Within that folder, you can create other directories to organize all graphics. Create one folder for site images and other folders for the databases that refer to images. ADS will provide further instructions on how to best coordinate images on the server and provide suggestions on automating the process of comparing image names stored in the database with what is actually in the folder.

Recommendations

- When creating database, layout, relationship, or field names do not use any symbols or spaces in the name. Avoid any non-alphanumeric characters (& $ # . ! : ; , < > ? / | \ - and so on). Make double sure that the name does not have a space at the end.
- Do not use ANY container fields.
- Use the ".fp3" extension at the end of all database names.
- Most of the time you should use relationship names that are the name of the database they are related to, but without the ".fp3" extension
- Give the primary layout that is referenced for the web the name "web" other layouts can have any names (as long as they do not use any spaces or non-alphanumerical characters). Make sure that no layout has the same name as the database.
- Web Page Integration
- Integration of databases with the HTML web files will be accomplished using Lasso 3.6.5 and JavaScript.

Server Configuration

In order to allow the web server to work at optimum speed, using an ISP who provides hosting services on a shared server.The web site HTML files and databases need to be served on a co-located server at some web hosting company. In this scenario, the client would purchase a computer, install all required software, configure the server, and then maintain day-to-day operations using remote software. The ISP would ensure that power and Internet access are always available to the server and that it is placed in a secured location. By co-locating their web server(s), the client will retain complete control over the serving environment and not be interrupted by high traffic or intensive FileMaker Pro database activity from other web sites on the shared server. One alternative to co-location would be to set up the web servers on a high-speed network within the client's office.

A list of proposed resources needed for a co-located server at a minimum would include:

- Macintosh computer (preferably G3/G4 CPU)
- Mac OS
- Web server application (Apache)
- FTP server application (Apache)
- Lasso
- Remote Reboot (hardware and software).
- 1 - phone line (to request reboot of server). Note some ISP may have a telnet option.
- Timbuktu Pro
- FileMaker Pro

The ideal setup for Lasso though is to have two computers and to use Lasso's FM Remote module to communicate with the FileMaker Pro databases on the second computer. In the two computer scenario, one computer would run the Web server, FTP server, and Lasso (and house the site HMTL format files), while the second would run FileMaker Pro. Both computers would need to have remote administration capabilities using Timbuktu Pro and the tools required to remotely reboot the server in case the computer crashes. The better of the two computers should be used to run your FileMaker Pro databases. A list of proposed resources, and the quantity of each, include:

- 2 - Macintosh computers (preferably G3/G4 CPU)
- 2 - Mac OS 8.5.1 (or higher)
- 1 - Web Server application (Apache)
- 1 - FTP Server application (WebSTAR 4 includes this, for separate software consider Rumpus)
- 1 - Lasso 3.6.5 Plug-in
- 2 - Remote Reboot (hardware, and software)
- 1 - phone line (to request reboot of server). Note some ISP may have a telnet option.
- 2 - Timbuktu Pro (at least 4.8)
- FileMaker Pro Unlimited 5.0

Further Web server details are yet to be determined. The client may want to consider an additional server for static HTML and images, in other words all pages not processed by Lasso for web database integration. Note this could be a co-located UNIX or Windows NT server, or the files could be uploaded to a shared web server at an ISP.

Installation
Details of server installation, configuration, and maintenance are unknown. These roles are to be defined in a future revision of this draft. There will be an initial period of time in which a test server is setup and software installed before making the solution available to users. During this time ADS will provide modifications of code to adjust the files to the new server location. In addition, a period of time should be set aside for client review and refinement of final solution.

Lasso
Lasso must be installed with (at least) the following modules:

- Lasso_Tags.mod
- Admin.mod (can be removed after the Lasso Administrator password has been set)
- FM_DataSource.mod
- Java_Adapter.mod
- Log_Tag.mod
- Macro.mod
- Math_Tags.mod
- Security_DataSource.mod
- String_Tags.mod

Before using the solution, Lasso Security must be set up. Open the configuration web page for Lasso Security using http://<host name>/Lasso/Security/. Add the following authentication information for all databases, and indicate that the following provide access privileges for search, add, delete, and update: Username: Artist_Workshop, Password: arti6414A. Any other security settings for Lasso are to be configured at the time of installation.

Shopping Cart Process
The basic concept of a "shopping cart" approach is as follows:

Step 1: Select items to purchase or training to register for.
Items can be added to a client's "shopping cart" at any point there is an item to purchase. Multiple items can be accommodated with a quantity text box. This data is stored in an orders database on the server. When an item is added a session ID is set and stored in both a cookie and passed throughout the web site using tokens (this requires all links in the web page to utilize Lasso). This session ID is used to identify the customer.
The shopping cart can be recalled at any point by the user selecting a "shopping cart" link that recalls the information (retrieved by referring to the current users session ID). The shopping cart allows the user to edit quantities or to delete items.

Step 2: Enter contact info on order form
The complete order form is filled out to gather needed details about the customer. The customer should specify the list of contact details they want to include on the invoice and which items are to be marked required. The order form includes everything but payment information. For example:

- First and Last Name (required)
- Address, City, State, Postal Code (a physical address is not required since the transfer of products is electronic)
- Email (required so that they can receive a response)
- Phone and FAX

When the order form is submitted JavaScript is used to verify that all required fields have been completed and that the email address is present and in an acceptable format. If there is a problem alert messages inform the customer to complete the page correctly. If all is complete the order is passed to the invoice page. The order form is to be stored in a database at this point. This order is not considered final until the payment process is complete.

Step 3: Invoice with credit card processing
The invoice page should be located on a secure server. The page would show the customer's contact info on the top and an itemized list of purchases with a total on the bottom. There would be two links to lead back to the order form or shopping cart page in order to edit one or the other. In addition, there would be a credit card form to pay for the invoice, the customer enters CC and date and type of card (give a limited choice of types). When this page is submitted, the credit card is verified and the order information is stored in a FileMaker Pro database.

Step 4: Receive Confirmation of Purchase
The reply page for submitting an order should show a confirmation message, the complete invoice once again (with no links to change it), and some sort of serial number that identifies the transaction. If a page is complex or shows a lot or surrounding graphics (or especially if its in a frame), provide a "Print Preview" link to open a new window from which the invoice can be printed. The session ID is erased from the system.

Programming Specifications and Development Time Estimate
Detail the various components of the site, and assign general time estimates to each item

Min Hours	***Max Hours***	***Functional area description***
5	20	Milestone 2 Some modifications will be necessary to the databases for web compatibility. Also, removal of embedded graphics and container fields will need to be corrected. Some additional fields will be required to contain information regarding user, as well as some site data that should be retained. The serial number scheme will need some modifications for security issues.
15	25	Milestone 3 Lasso and HTML integration with databases for product search, navigate and display.
15	25	Milestone 4 Lasso and HTML integration with databases for Cart display/add/update/delete. JavaScript verification for update and delete.
20	40	Milestone 5 Lasso and HTML integration with databases for Order completion, saving user information. Integration with secure payment server. This may take longer if integration needs to be done using Java, may have to write a custom Lasso Java Plug-in to interface with secure server.
5	15	Milestone 6 Additional debugging and security testing, checking for security issues and eliminating problems.
60	125	**TOTAL**