# Online Image Retail Management Site

Project Documentation

By: Nauraj Pannu, Richard Thomas, Paul Morrill

Date: December 1st , 2008

Richard Thomas, Paul Morrill, Nauraj Pannu

Project Statement

The Title:

Online Photography Image Retail System

The Project:

A web-based application will attempt to solve the problems that photographers face when they try to sell images over the Internet. The project will accomplish this by a series of simple to use web pages, for both the photographer and the customers.

The Problem:

Photographers face a unique problem when they attempt to sell their images over the web. The problem lies in that there are no easy to use systems to accomplish this task. For a photographer to be able to sell their images online they must have an in depth knowledge of web design and server side scripting, this is the problem because the average photographer does not have this knowledge.

The Impact:

The Online Photography Image Retail System will make it simple for the average photographer to have a great online retail image store. This will allow them to compete for other larger photography companies that have their own in house systems. The photographer will be able to get their images online and ready to sell in a short period of time by using simple ftp software and a few short administration web pages. This will allow the small photographers out there to focus on what's important, the photography.

The Motivation:

Having been a systems administrator for a large wedding photography company for three years, I can see the clear demand for a system like this. From dealing with the photographers personally from day to day the need for a simple web based solution stands out. This system would be a time saver and efficiency booster for many small photography companies and individual photographers.

Detailed Outline of Individual Components:

Functions for this project are not limited to the application of images only. They are purposely made to be easily applied to other merchandise. This was done with the intent to allow the system to have features added without a large over head.

Components of the Admin Web site:

- Event Manager: Provides group and event setup to both the super administrator and the individual group administrators.
    - Functions of the event manager:
        - For a list of specific functions implemented so far see diagram 2.1
        - List/Add/Edit/Delete Master Groups
            - These function allow the super administrator to list all of the master groups, create a new master group by using add, edit an existing master group, and delete and existing master group.
            - There will also be functions to manage the specific attributes of each master group; they will be defined at a later date.
        - List/Add/Edit/Delete Sub-Groups
            - These functions allow the super administrator to list all of the sub-groups, create a new sub-group by using the add function, edit an existing sub-group with the edit function, or delete an existing sub-group using the delete function.
            - There will also be function to manage the attributes of each sub-group they will be defined at a later date.
        - List/Add/Edit/Delete Events
            - These functions allow both the super administrator and the individual sub-group administrators to list the events of a specific sub-group, to create a new event for a specific sub-group using the add function, to edit an existing sub-group event by using the edit function, or to delete an existing sub-group event by using the delete function. The sub-group administrators can only use these functions on sub-groups that they are a part of. The super administrator may execute these functions on any event under any sub-group.
            - Manage attributes
                - Allows the sub-group administrator to specify specific details about each event. The fields that will be defined are: event name, date of the event, time the event is to be held, who is working the event, if the event has a special event image, where the event is to be held, etc…..

- Manage Images
    - This function will allow both the super administrator and the sub-group administrator to remove images that they do not want displayed for a certain event. Again the sub-group administrators are limited to only removing images from events under their sub-group, while the super administrator can remove images from any event across every sub-group.
- Manage Guest Sub-Groups
    - This function will allow both super administrators and sub-group administrators to specify if any other sub-groups are related to an event within their own sub-group.

- Event Search: This function will allow both the super administrator and the sub-group administrators to search the system for a specific event. For the super administrator it will search all events, and for the sub-group administrators it will only search the events for which their specific sub-group is a part of.
- Event Calendar: This function will allow a calendar of events to be viewed. It will be both global for the super administrator and specific to each sub-group for sub-group administrators. It will also allow the system to notify the correct parties when an event is coming up, and if a specific amount of time has passed without the system receiving the images.

- Customer Manager: Provides the ability for the super administrator and the sub-group administrator to manage members of each sub-group's attributes.
    - Functions of the Customer Manager:
        - Customer Search: Allows the super administrator to search the entire system for a certain customer. Allows the sub-group administrators to search for a customer within their specific sub-group.
        - Add/Edit/Delete Customers
            - This allows the super administrator to create a new customer, edit an existing customer, or delete an existing customer, throughout the entire system.
            - This allows the sub-group administrators to create a new customer for their specific sub-group, edit a customer of their specific sub-group, and delete an existing customer of their specific sub-group.
        - Manage Customer Information
            - Allows the super administrator to manage customer information across the entire system.
            - Allows the sub-group administrators to manage customer information on their specific sub-groups only
        - List Orders

        - Allows the super administrator to list the orders and the order details for each sub-group. The super administrator may edit order details and change the order status.
        - Allows the sub-group administrators to see a list of completed and pending orders for their specific sub-group
    - Production Manager: Allows the super administrator to view the orders that are pending being printed
        - Functions of the production manager:
            - List pending orders
                - Lists all orders that are pending being printed
            - Render packing slips
                - Creates and send to the printer packing slips for a specific batch of orders, a single order, or multiple selected orders
            - Order Details
                - Shows the complete details of an order, and allows the super administrator to manually change the status of an order
            - Search Orders
                - Allows the super administrator to search the entire system for specific orders.
    - Management of Ads and Sub-Group Images: Allows the super administrator and the sub-group administrators to manage what ads are seen and what images are used as the sub-group icons.
        - Functions of the ads and sub-group image manager
            - Sub-group image
                - The image displayed next to the name of the sub-group on the user page. Allows the image to be changed or edited
            - Ads
                - Allows the administrator to specify what ads are shown to the sub-group specific customers.

Components of the Asset Management System

- Thumbnail Sync
    - Process where new thumb nails are created from newly upload event images. After the images are uploaded to the system, the system registers the frames, creates the thumbnails, and then pushes the complete batch to the website.
- Full Image Sync
    - Process where the full sized images are sent to the printer's hot folder. The full sized images are pulled for the selected thumbnails in the order, they are then copied to a temporary file that is resized and cropped to fit the specific order, this file along

with the order details is then placed in the printers hot folder, and the pending order list is updated.

Components of the Retail Site (User Interface)

- This is the front end for the customers to access their events and images.
- Process for front end
    - Choose a master group ( e.g., Louisiana)
    - Choose a sub-group ( e.g., Jane Smith)
    - Choose an event ( e.g., Wedding)
    - The system then list's the images in the event and shows the allowed options
        - The allowed options are
            - Allow direct download of the image
            - Add the image to the cart
            - Set a custom crop on the image
            - Select multiple images to add to the cart
            - Get a large preview of the image, or images
    - View Cart
        - Lists the contents of the specific customers cart
        - Allows the customer to continue shopping, edit the quantity of the pictures they would like to order, purchase a digital copy if offered, and remove items from their cart.
        - Allows the customer to proceed to the check out process
    - Check out process
        - Allows the customer to specify the billing address, the shipping address, the payment option they would like to use, and if they would like an email confirmation with the receipt attached
        - Creates a new order and adds it to the list of pending orders in the system, sets the status of the payment, and sets the complete order details
        - Generates emails for the administrators and the customer (if the option was selected) to confirm that the status of the payment.

Components of the system

- Database design
    - The database is the main component of this program.
    - It is SQL Server from Microsoft
    - For the specific tables and their specific rows and columns refer to diagram 1.1 attached at the end of this document.
- Language
    - This system will be written in .net
    - This will make it easy to set up and deploy regardless of the onsite computer assets.

    - This will allow the system to be set up and managed on a central server and to be used remotely over the internet, or a company intranet
- Server properties
    - Dual core AMD Athlon X2  64 2.8Ghz
    - 2GB of RAM
    - 500GB Hard drives in a raid 1
    - 10Mbps internet connection up and 2Mbps internet connection down

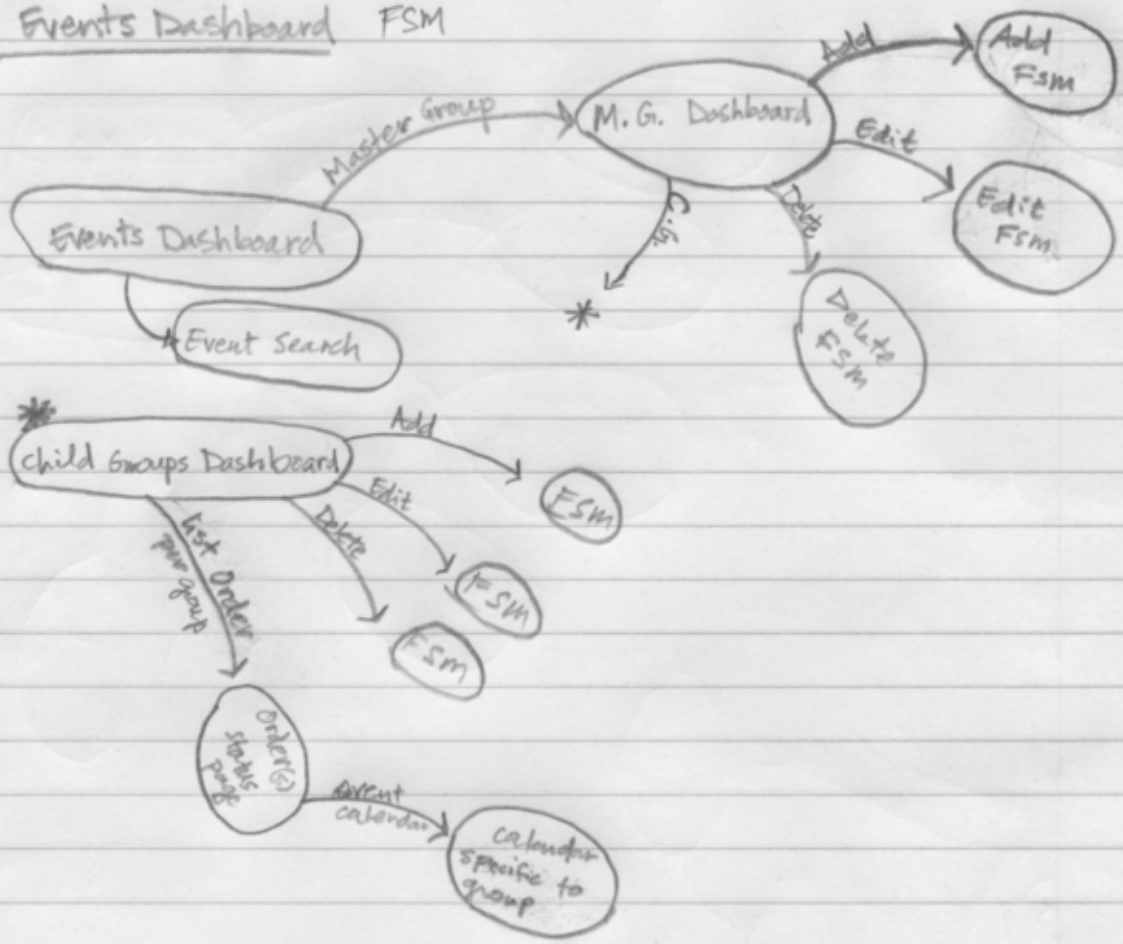
Common terms used in the outline:

- **Hot folder:** The folder on the print server where images that are ready to be printed are placed. Once they are inside the folder the printer will automatically begin printing them.
- **Print Server:** The computer server that runs the software required to operate the commercial photo printer, or printers.
- **Event:** These are specific things that take place on a given day at a given time. Examples of events are weddings, private parties, fraternity parties, and school dances. The events are where the photos are placed. An example would be Jane Smith's wedding photos placed under the event wedding, which is then in turn placed under the sub-group Jane Smith, which is then placed under the master group weddings.
- **Master Group:** This is the main set of categories that help to separate the different sub-groups. Examples of the master groups would be LSU, Weddings and Private Parties.
- **Sub-group:** These are the categories underneath the master groups. They contain each specific event for the sub-group. An example would be the sub-group Jane Smith under the master group wedding, with the events of wedding, rehearsal dinner, and bridal portraits.
- **Digital Copy:** This is the image file in a .jpeg format. It will come complete and cropped with a title.

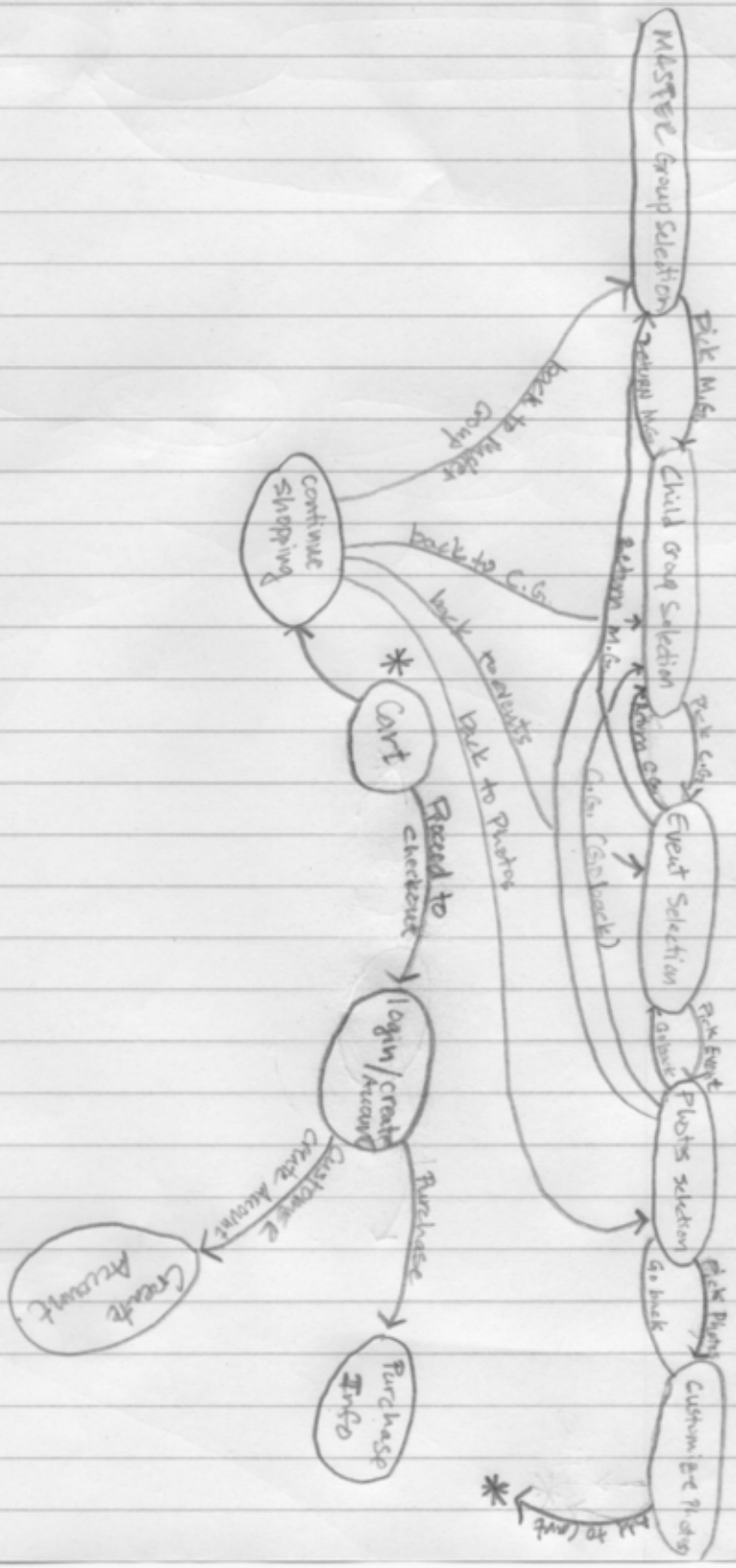
# Admin Site FSM

(All nodes can return or access each other.)

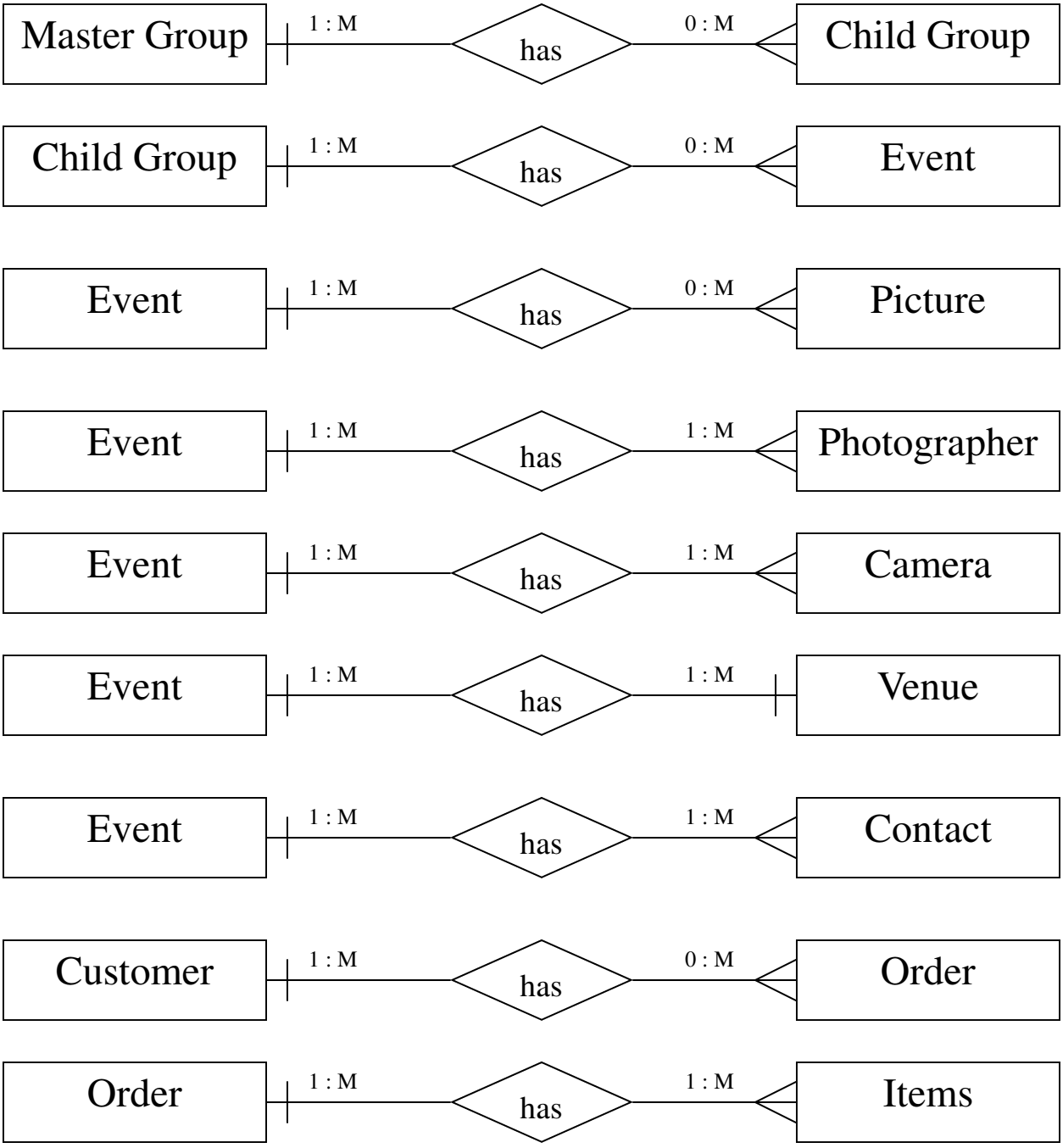
Events Dashboard FSM



CONSUMER USER INTERFACE FSM



# ER Models


Master Group
1 : M
has
0 : M
Child Group
Child Group
1 : M
has
0 : M
Event
Event
1 : M
has
0 : M
Picture
Event
1 : M
has
1 : M
Photographer
Event
1 : M
has
1 : M
Camera
Event
1 : M
has
1 : M
Venue
Event
1 : M
has
1 : M
Contact
Customer
1 : M
has
0 : M
Order
Order
1 : M
has
1 : M
Items

# Functions :Admin Site

Diagram 2.1

Events Manager

{

_Functions: Master Group
- _List
- _Add
- _Edit
- _Delete

_Functions: Child Group
- _List
- _Add
- _Edit
- _Delete

_List Orders Per Group

_Event Search
- _EventCal
  - _GroupSpecificCal

}

Global Functions

{

_List Orders
- _complete
- _pending

_Ad-Setup
- _List
- _Edit
- _Add
- _Delete
- _ActivateDeactivate
- _Timer

_Logo
- _Add
- _Delete
- _Edit
- _List

}

Production Manager

{

_ListPendingOrder
_RenderPackingSlip
_OrderDetails
- _GetOrderAttributes
- _SetOrderStatus
- _EditOrderAttributes
- _SearchOrder

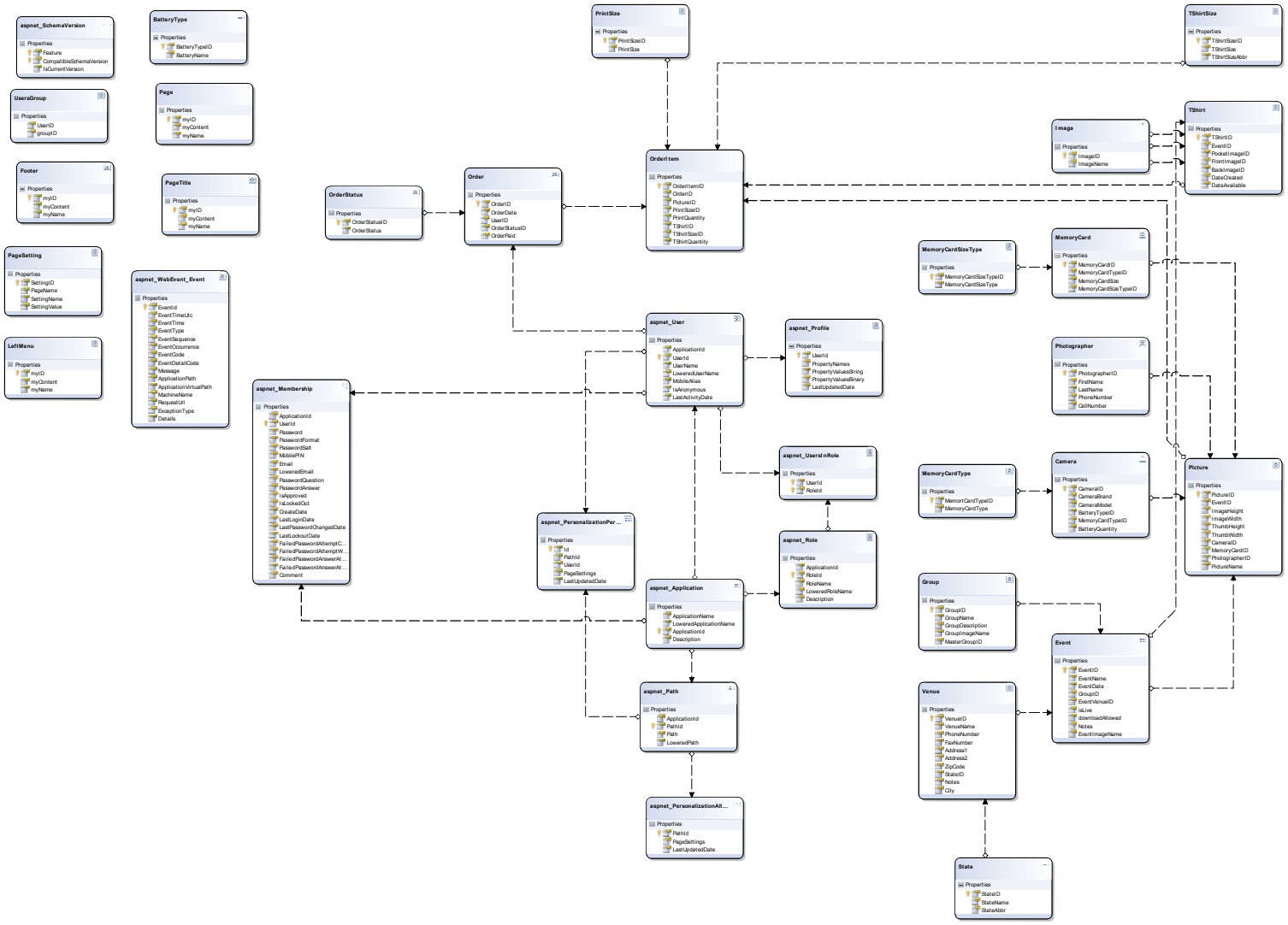
}

Customer Manager

{

_CustSearch
_Add
_Edit
_Delete
_List
_ListCustOrder

}


aspnet_SchemaVersion
Properties
Feature
CompatibleSchemaVersion
IsCurrentVersion
BatteryType
Properties
BatteryTypeID
BatteryName
PrintSize
Properties
PrintSizeID
PrintSize
TShirtSize
Properties
TShirtSizeID
TShirtSize
TShirtSizeAbbr
UsersGroup
Properties
UserID
groupID
Page
Properties
myID
myContent
myName
Image
Properties
ImageID
ImageName
TShirt
Properties
TShirtID
EventID
PocketImageID
FrontImageID
BackImageID
DateCreated
DateAvailable
Footer
Properties
myID
myContent
myName
PageTitle
Properties
myID
myContent
myName
OrderStatus
Properties
OrderStatusID
OrderStatus
Order
Properties
OrderID
OrderDate
UserID
OrderStatusID
OrderPaid
OrderItem
Properties
OrderItemID
OrderID
PictureID
PrintSizeID
PrintQuantity
TShirtID
TShirtSizeID
TShirtQuantity
MemoryCardSizeType
Properties
MemoryCardSizeTypeID
MemoryCardSizeType
MemoryCard
Properties
MemoryCardID
MemoryCardTypeID
MemoryCardSize
MemoryCardSizeTypeID
PageSetting
Properties
SettingID
PageName
SettingName
SettingValue
aspnet_WebEvent_Event
Properties
EventId
EventTimeUtc
EventTime
EventType
EventSequence
EventOccurrence
EventCode
EventDetailCode
Message
ApplicationPath
ApplicationVirtualPath
MachineName
RequestUrl
ExceptionType
Details
aspnet_User
Properties
ApplicationId
UserId
UserName
LoweredUserName
MobileAlias
IsAnonymous
LastActivityDate
aspnet_Profile
Properties
UserId
PropertyNames
PropertyValuesString
PropertyValuesBinary
LastUpdatedDate
Photographer
Properties
PhotographerID
FirstName
LastName
PhoneNumber
CellNumber
LeftMenu
Properties
myID
myContent
myName
aspnet_Membership
Properties
ApplicationId
UserId
Password
PasswordFormat
PasswordSalt
MobilePIN
Email
LoweredEmail
PasswordQuestion
PasswordAnswer
IsApproved
IsLockedOut
CreateDate
LastLoginDate
LastPasswordChangedDate
LastLockoutDate
FailedPasswordAttemptC...
FailedPasswordAttemptW...
FailedPasswordAnswerAt...
FailedPasswordAnswerAt...
Comment
aspnet_UsersInRole
Properties
UserId
RoleId
MemoryCardType
Properties
MemoryCardTypeID
MemoryCardType
Camera
Properties
CameraID
CameraBrand
CameraModel
BatteryTypeID
MemoryCardTypeID
BatteryQuantity
Picture
Properties
PictureID
EventID
ImageHeight
ImageWidth
ThumbHeight
ThumbWidth
CameraID
MemoryCardID
PhotographerID
PictureName
aspnet_PersonalizationPer...
Properties
Id
PathId
UserId
PageSettings
LastUpdatedDate
aspnet_Role
Properties
ApplicationId
RoleId
RoleName
LoweredRoleName
Description
Group
Properties
GroupID
GroupName
GroupDescription
GroupImageName
MasterGroupID
aspnet_Application
Properties
ApplicationName
LoweredApplicationName
ApplicationId
Description
Event
Properties
EventID
EventName
EventDate
GroupID
EventVenueID
isLive
downloadAllowed
Notes
EventImageName
aspnet_Path
Properties
ApplicationId
PathId
Path
LoweredPath
Venue
Properties
VenueID
VenueName
PhoneNumber
FaxNumber
Address1
Address2
ZipCode
StateID
Notes
City
aspnet_PersonalizationAll...
Properties
PathId
PageSettings
LastUpdatedDate
State
Properties
StateID
StateName
StateAbbr

# Initial Projections for Online Image Retail Management Site

Initially, this project had a grand spectrum of ideas and was going to be huge in detail. The site was going to be a major online photo shop with all the bells and whistles like having the whole site run directly from client side with the engine being quiet on the database side. This was going to work with the upload option working similar to Facebook's image upload option where you could go online, log in, and upload the pictures client side without the need of having separate connecting software developed to upload the images. Implementation of a database was necessary to host all the items and tie them together so that they would run smoothly when the site was generated. This was so the site was dynamic. Intuitively, the admin side was going to be a dashboard with access to every part of the website and every function built-in to handle event, groups, users, photographers, cameras, memory cards, etc. The admin could generate reports status of orders, how many pictures had been ordered from specific events, who are the different groups in each master group and which photographers are tied to those groups along with camera details and such. The client would have the option of being able to go to the event, then login and select the photos depending on her connection to the photos and its events. The photographer would give access to his or her photos through a password and could give this out to the photographer's clients. This would give them access to those images only and would help secure the images also. The site was to have a very graphically embedded design in that it was extremely user friendly and could be manipulated to the users choice and setting for that user only. The pages were to run off of one master page and all of the content would be embedded in dynamic placeholders in certain sections of the actual website. The images that were to be selected were going to be done so by highlighting them and then adding them to a cart. The images in the cart were then to be customized by size, picture, quality, paper type, etc. The site would have a checkout feature where the customer could buy the images they customized and have they delivered to their doorstep. The design of the page was to be dynamic as said earlier but with everything working off of a master page. The images were to be dynamically changing in that they would be easily manipulated when they were selected and it would happen independently of the rest of the site.

# Actual Implementation of Online Image Retail Management Site

When implementing the site, we ran into problems that hindered our ability to create our projected implementations for this project. Mainly we ran into the issues of getting what we wanted to worked together, to actually work together. What we did get working was simply what happened to work efficiently. For instance, a customer or user could login to the site initially and depending on their security clearance, it would give them access to whatever they could work with. If they have admin capabilities, then they could access the admin side of the site, otherwise they could go straight to the pictures. Going to the admin side, we have given the admin capabilities of manipulating the site by having the ability to add events, users, photographers, and groups. Unfortunately we were unable to give the admin the access to print reports and handle the reporting part of the dashboard. Concurrently, the admin doesn't have access to orders and production either because this was a part that we didn't need to develop. For the uploading and manipulating of the images feature, we developed our very own FTP Client that would allow the photographer to upload his images specific to the events and send them to the server with the address in the database giving it the ability to become live once uploaded. The images are also automatically resized and thumb nailed to fit comfortably on the website and not mess up the site smooth flow. With the ftp client, it defeated the purpose of creating a site direct uploading feature that would have been extremely intense in design and implementation. The client side was handled almost concurrently with the predicted outline but had a few quirks. One in particular was the ability to highlight an image and add it to the cart. The site actually allows a user to check the image through a check box which can then be added to the cart. The database was handled exactly how we foresaw it to be since there was no trouble creating this key part of the project. The main issues we had were developing the queries to pull information from the database and display it in content holders on the master page to give a website. This is what took the longest time to do.

# Implementation of Components

Based on Previous documents, we described in detail a few of the main components of our project and what they would consist of in approach and task. To emphasize on the approach and task, the components will briefly be explained:

Administration:

The Admin management is all done server database side with a visual client through the website securely. With the administration part handled by one person, they would have access to the critical parts of the Management Console. This includes access to the database and the security clearance to manipulate the database with strict functions built into the console. Another aspect is handling the customer, or user, membership side where they would be added to the system through forms built-in to the site. This is looked over by the admin and then once this is done, in theory, the admin would handle the orders by submitting them as a customer order (This part was not finished because it would have required payment options and such. Dr. Kundu stated that this was not required in the time that we had to finish.) Along with user functions, there are photographer, event, and group functions. The event functions are handled by generating an event ID and allocating it to such group depending on how it was created. Groups are created the same way. The additions of a photographer are done by the admin also once a request is submitted. This tied into the FTP Client that we built, where the Photographer has a chance to upload his Images to the Database and when this is done, The photos are resized and thumb nailed automatically ( also built in function that were created by us).

Client:

Basically this part is simple (in theory). The client side is the website part which the users are able to see and handle. Once they get to the site they are given the opportunity to submit request and contact Photography Shoppe or create a new user. The other thing they can do is log in. This gives them access to the photos. They would then choose a Master Group (Weddings, University, High School, Sports, and Private Parties) where they would be led to the Sub-group and given the chance to choose the next level. Then they choose the event and the pictures are pulled from the Database. They are then allowed to add them to cart by checking off the box right under an image they would want. This is the basic concept of our site, Image Retail Management.

The Project was broken up into 4 major components: Client Side Design and Implementation, Website Design and Implementation, Administration Design and Implementation, and Database Design and Implementation.

Client side was the photo management with the events and groups tied based on the configuration of them individually in the database.

Website side was basic in that it was a Master Page with style sheets and also grabbing from the Database all of its content based on queries through the system. The major component of the Website was the separation implementing of the database. Once this started to work it was great in that they site ran smoothly.

Administration side was the main part of the site since it handled all the components of groups, events, and photo implementation. As stated previously in this document, one can see that this block of design and implementation took the longest. This part was closely connected to the Database side.

The Database was tricky because all the relationships had to be proper. The Database held all for the information that fueled the whole site. This part was simple once we got it to work and really makes the website. The images are all “logged” into the database and are then grabbed and placed on the site. The photos are grabbed based on the groups and events. The coding of this was not just the database alone but was ran by creating queries that took parts of the database and user request and grabbed the photos and events from the different constructs that were created through the database.

FTP Client was developed to feed the database with images that the photographer takes at specific events. This tool is may be the greatest part because it gives the photographers a feel of just how simple the site will be and how easy it would be for them to use this site.

In all the Website, or Management Console, seems to work around the Database and with the FTP Client giving the database and the website its images, the site as a whole can run and do what it is intended to do.

# Full Walkthrough of Online Image Management Retail Site

Once you access the site, you have different options. As a new user, or customer, you can request membership. This is done by submitting the request to become a new user. After this is taken care of, you can log in. The site then takes you to the master group page where you can select the master group of Weddings, University, High School, Sports, and Private Parties. After you have selected this option you are taken to the master group's subgroups where you can select one of the subgroup depending on which master group was selected. For weddings, it is simple in that you select the subgroup based on the couple's names. For university, it could be a Greek organization to a academic function. High school subgroups could be dances to senior portraits and so on. The next page would be specific events for the subgroup. Then you care taken to the images. You can select the images by checking the check box below each image. For full functionality purposes, you could then add them to the cart and request an order for the images you would like to order. You could then customize the images by selecting options available.

As a photographer, you would request access. Once this is granted, you could then start uploading the images by using the FTP Client. The client asked you to present an Event ID and a user name and password. Then you browse for the images you want to upload and the images are uploaded to the event based on the Event ID given. The images automatically become live and are resized and thumb nailed to fit properly on the website and look uniform.

As an Admin, you would access the page through an admin restricted area. This side is where you could add events, users, photographers, and groups. You have the ability to directly affect content.

# Testing and how it was done

The website was tested through many different trials throughout implementation. With the development of each component, the testing took place to see if the result was what was expected. From the standpoint of FTP Client, testing was very necessary because this would be a crucial part of the whole project. When the interface was working, we tested to see what it would do by inputting data. The first time nothing happen because it wasn't tied to the database. After some manipulation to the code and directing it through the ports over the network, the images uploaded successfully without hesitation. The code was efficient enough to let it do and worked after many trials. The interface even lets you know just how much of the uploading has completed because it has a progress bar built-in to it. When the database was connected through the different forms or adding groups, users, event, and photographers, we needed to see if the additions were getting added to the database and they were. The inputs were done through the website and then once we had submitted it, we could see that the database had been modified and add new data to it by running queries on the database. This was great because with the database and the ftp client working the battle was half way done. The next part was grabbing the information off the database and displaying it on the website. This was also done through query functions from the code which called the database and located the specific data. Depending on what category the images were, they showed up in the corresponding sections of the website but only after many trials and many errors. We at first didn't get any information to display. The problem was definitely on how it was coded to grab the information. We manipulated the code once again and saw that things were getting better. Some of the information was there but the pictures weren't. So we went ahead and saw that we had called the function incorrectly and finally fixed the error. This was the problem because it was showing the placeholder for images but no images. This also fixed the issue we had with the names of the events not showing up and how they were displaying. The main problem that we had was with the queries and how they were called from the database.