CS 148 Database Design for the Web

Doug Krieger

Final Assignment

Emily Deutchman's Art

Version <1.0>

|  |  |  |  |
| --- | --- | --- | --- |
| Time Log | | | |
| Date | Time Spent (in hours) | Description | Author |
| 11/20/13 | 3.0 | First Version of Software Requirements | Doug Krieger |
| 11/20/13 | .25 | Estimated time to complete assignment \_\_\_\_48\_\_\_ hours | Doug Krieger |
| 11/27/13  - 12/13/13 | 96 | Complete Website | Doug Krieger |
| 12/13/13 | .25 | Update Project Specification (v2) | Doug Krieger |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table Of Contents

Software Requirements Specifications 4

Introduction 4

Purpose 4

Definitions, acronyms, and abbreviations 4

Overview 4

Overall Description 5

Data Dictionary 5

E-R Diagram 5

Schema 6

Story Board 7

Specific requirements 7-8

# Software Requirements Specifications

## Introduction

### Purpose

The purpose of this document is to describe the requirement specification for the web site “Emily Deutchman's Art”.

The intended audience for this document is the prospective developers of this site. The goal being that you can give this document to someone and they can make the site without having to ask any questions.

### Scope

Create a website with a simple GUI to add, delete, update, and (most importantly) display artwork. It should give pricing and a way to contact the admin/artist. A system of email notifications must be implemented so the admin has record of content changes. A tag system must be implemented that allows visitors and the admin to browse the gallery by category. All of this information must be stored in a database.

### Definitions, acronyms, and abbreviations

HTML – Hypertext markup language – used to define your content.

PHP – Personal Home Page – language that helps to customize html.

CSS – Cascading Style Sheets – used to define the look of a web site.

W3 Validation – refers to both Html and CSS validation tool provided by the W3c.org. the html validator is located at:

<http://validator.w3.org/>

with the CSS validator located at:

<http://jigsaw.w3.org/css-validator/>

### 

### Overview

## Overall Description

The site is about efficiently managing an art portfolio on the Internet without any coding knowledge. As requested by Emily, the styling is very minimal and will be progressively enhanced over time to her liking. Prices are displayed, but it is up to the visitor to email the admin to make inquiries. This is intentionally not a fully automated ecommerce site, as Emily feet that would detract from the artistic value.

Emily Deutchman's Art allows the site manager (the artist) to put her artwork on the web with a simple, clean interface. Art can be categorized with any number of tags and website visitors can browse filtered by tag or unfiltered. The admin portal allows the adding, deleting, and editing of content in the same location it appears. The admin(s) can toggle between admin view and standard view seamlessly. Emails are sent when records are updated, and visitors see a link on each item's page to email the administrator with the item's ID in the subject.

### Data Dictionary

Data Dictionary for database `DRKRIEGE\_assignment7.1`||

=====================================================||

mysql> describe tbl\_art;

+-------------------+--------------+------+-----+-------------------+-----------------------------+

| Field | Type | Null | Key | Default | Extra |

+-------------------+--------------+------+-----+-------------------+-----------------------------+

| pk\_art\_id | int(11) | NO | PRI | NULL | auto\_increment |

| fld\_display | tinyint(4) | YES | | NULL | |

| fld\_name | varchar(50) | NO | | NULL | |

| fld\_img\_src | varchar(100) | NO | UNI | NULL | |

| fld\_description | varchar(200) | YES | | NULL | |

| fld\_availability | tinyint(4) | NO | | NULL | |

| fld\_price | smallint(6) | YES | | NULL | |

| fld\_last\_modified | timestamp | NO | | CURRENT\_TIMESTAMP | on update CURRENT\_TIMESTAMP |

+-------------------+--------------+------+-----+-------------------+-----------------------------+

8 rows in set (0.00 sec)

mysql> describe tbl\_tag;

+-------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-------------+-------------+------+-----+---------+-------+

| pk\_tag\_name | varchar(50) | NO | PRI | NULL | |

+-------------+-------------+------+-----+---------+-------+

1 row in set (0.00 sec)

mysql> describe tbl\_art\_tag;

+-------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-------------+-------------+------+-----+---------+-------+

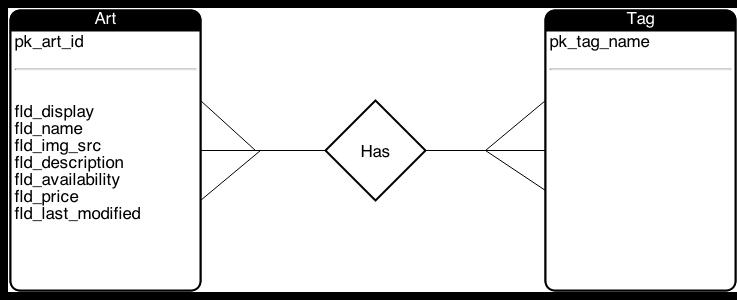
| fk\_art\_id | int(11) | NO | PRI | NULL | |

| fk\_tag\_name | varchar(50) | NO | PRI | NULL | |

+-------------+-------------+------+-----+---------+-------+

2 rows in set (0.00 sec)

### E-R Diagram



### 

### Schema

create table tbl\_art (

pk\_art\_id int primary key auto\_increment,

fld\_display tinyint,

fld\_name varchar(50) not null,

fld\_img\_src varchar(100) not null,

fld\_description varchar(200),

fld\_availability tinyint,

fld\_price smallint,

fld\_last\_modified timestamp,

unique index (fld\_img\_src)

);

create table tbl\_tag (

pk\_tag\_name varchar(50) primary key

);

create table tbl\_art\_tag (

fk\_art\_id int not null,

fk\_tag\_name varchar(50) not null,

primary key (fk\_art\_id, fk\_tag\_name),

foreign key (fk\_art\_id) references tbl\_art (pk\_art\_id),

foreign key (fk\_tag\_name) references tbl\_tag (pk\_tag\_name)

);

### Story Board finalStoryBoard.jpg

## Specific requirements

Here is just a list of things I will require you really need to write the specific requirements as they relate to your project.

1. *Location* – all files will be located in the public folder of your uvm account. The assignment web page and the submit process will detail the exact location. Failure to have your site located in the correct folder by the due dates will result in a zero on the assignment. Be sure to pay attention to the lowercase letters.
2. *Html Validation* – All pages will pass W3C Html validation for html 5.
3. *CSS Validation* – All pages will pass W3C 3.0 CSS validation.
4. *Meta Information* – All pages will contain a proper title tag, Meta tags (author, character set and description).
5. *CSS* – all pages will have a linked style sheet.
6. *Navigation* – All pages will contain navigation to all other pages on the site using an ordered list. Be sure to enclose the navigation in the correct element.
7. *Content* – Each page will have a minimum of 150 words not counting titles, lists or links. Be sure to use the correct elements to hold your content.
8. *Browser compatibility* - This site will be checked on Firefox, Safari, Chrome and Internet Explorer.
9. *File Names* – the main home page will be called home.php with the rest of the file names up to you (be sure to use .php, .css for the respective files).

10. 3 tables: tbl\_art, tbl\_tag, tbl\_art\_tag

11. Interface must be easy

12. minimum 1 form, all forms must validate information/ give error info

13. form must allow add/delete/update for all tables

14. automatically delete relational records

15. pages must be valid html/css

16. upon completion of form email is sent. user clicks link, database reflects confirmation

17. must use git/ github