|  |
| --- |
| CS 5338 (Database Design) w/ Dr. Hwang, Fall 2009, Texas State University |
| Online Musical Performance Submission and Rating Website |
| Milestone 1 |
|  |
| **Virpobe Paireepinart** |
| **10/20/2009** |

|  |
| --- |
| This document contains all of the deliverables for the database design from inception to midterm. |

Table of Contents

[Introduction 4](#_Toc243823182)

[Motivation 4](#_Toc243823183)

[Project Description / Scope 4](#_Toc243823184)

[The Software and Platform Used to Prepare This Document 6](#_Toc243823185)

[The Software and Platform Used to Implement This Application 6](#_Toc243823186)

[Study of System Functional Requirements 7](#_Toc243823187)

[Introduction 7](#_Toc243823188)

[Summary of Interview / Study 7](#_Toc243823189)

[Functional Requirements – Input Business Forms / Screens 8](#_Toc243823190)

[Registration For Site 8](#_Toc243823191)

[Logging Into the Site 9](#_Toc243823192)

[Login Details Recovery 9](#_Toc243823193)

[Submitting a Performance 10](#_Toc243823194)

[Rating a Performance 11](#_Toc243823195)

[Searching for Performances 12](#_Toc243823196)

[Assigning Reviewers to Performances 12](#_Toc243823197)

[Functional Requirements – Output Business Forms / Screens 13](#_Toc243823198)

[Viewing Reviews of Performances 13](#_Toc243823199)

[Reading Comments on Performance 13](#_Toc243823200)

[Viewing Performance Information 14](#_Toc243823201)

[Listening to a Performance 14](#_Toc243823202)

[External Schema (User’s Views) 15](#_Toc243823203)

[Diagram of System Model of the Project 15](#_Toc243823204)

[Tables before normalization 16](#_Toc243823205)

[User Table 16](#_Toc243823206)

[User Table (cont’d) 16](#_Toc243823207)

[Performance Table 16](#_Toc243823208)

[Performance Table (cont’d) 16](#_Toc243823209)

[Performance Table (cont’d) 17](#_Toc243823210)

[Normalized Relations 17](#_Toc243823211)

[Representing by Keys (Primary, Foreign; Optional Alternate or Secondary) 17](#_Toc243823212)

[Functional Dependencies of Every Normalized Relation 17](#_Toc243823213)

[Special Restrictions for Each Relation (optional) 17](#_Toc243823214)

[Conceptual Schema and Logical Model of the System 18](#_Toc243823215)

[Descriptions of All Required Relationships 18](#_Toc243823216)

[E-R Diagram of the System with Complete Tables 18](#_Toc243823217)

[Physical Constraints: Number of Occurrences, Frequencies of Report, Length of Lines, Response Time for Query, Response Time for Update Transactions, Security Constraints 18](#_Toc243823218)

[Queries and System Implementation I 19](#_Toc243823219)

[Twenty Nontrivial Queries in English with Some Complexity 19](#_Toc243823220)

[Queries the Administrator may want to do 19](#_Toc243823221)

[Queries a Reviewer may want to do 19](#_Toc243823222)

[Queries a Submitter may want to do 19](#_Toc243823223)

# Introduction

## Motivation

The idea behind this project was to provide a place where professors and other musicians could submit their performances to be peer-reviewed. Previous to this project, there is no existing place where performers can get peer-reviewed, which makes it difficult for universities to rate professors and for professors to have a large body of peer-reviewed work when they transfer universities. In other disciplines, professors must publish work in journals that is then peer-reviewed, but professors who specialized in musical performance had no way of displaying their unique talents and qualifications.

## Project Description / Scope

The only musicians and researchers who are able to submit anything to research journals are those who study the technical aspects of music, such as composition, meter, etc. Many musicians do not specialize in music technically, through analysis, but rather by performing it. These professors and other researchers who explore new ways of performance art are not able to make submissions to these journals, and thus there is no way to determine which performers have merit and which do not.

In this project, I will develop an online website. Professors or any other performance artists may come to the website to submit their work for peer review. They will have to provide detailed information about the performers of the piece, the origin of the work they're performing, and many other such details. Their submission will not immediately be visible to the public.

Then, the administrator will be able to access the website and log in. There will be other users as well, who are peer reviewers. The admin will assign 3 reviewers to each piece of music, and they will receive e-mail notification that they need to review it. Once they have reviewed it, the admin will get an e-mail that a review has finished. He must then go and determine if it was approved or rejected by each reviewer, and decide whether it is added to the collection of peer-reviewed submissions. If it is added, it is then viewable to the public.

The reviewers will be able to access the site and log in with their credentials, and then stream the music to their computer while they read over the technical notes the submitter included with their submission. They will then be able to vote whether the piece passes or fails the peer review, as well as giving comments regarding their decision that the submitter can read later.

## The Software and Platform Used to Prepare This Document

I used Microsoft Word 2007 to prepare the document, with help from Dia, a diagramming program, for the E-R model. Tables were built in Word as well.

## The Software and Platform Used to Implement This Application

Currently, none of the project has been implemented, so this section is To Be Determined. I will likely use Python for the server-side scripting, and MySQL for the database backend.

# Study of System Functional Requirements

## Introduction

In order to gather the functional requirements of the system, I had to conduct an interview with the client (in this case, my current boss.) I asked him questions to extract necessary information to form functional requirements of the project, and collected business forms for both input and output procedures that the website had to handle.

## Summary of Interview / Study

As the project that we are trying to accomplish doesn’t currently exist in any manner, all of the requirements had to be gathered by discussing with the professor how he would like the program to work. He detailed that he’d like to be an administrator, and that anyone who accessed the website would be a submitter, a submitter/rater, or a submitter/rater/administrator. If they were a submitter, they could only view other people’s submissions that had been approved, and submit their own submissions. If they were a rater, they could rate submissions that they had been assigned to as well. If they were an administrator, they could see all submissions currently on the system, and assign reviewers to specific submissions. He described the required information each user must have to register, as well as what information the users must submit with their performance so it can be properly cataloged for later retrieval.

## Functional Requirements – Input Business Forms / Screens

The actions a user takes when accessing the web site depend on the user's desired goal. If the user just wants to see all of the peer-reviewed and approved submissions in the database, the user does not need to complete either the registration or the login steps, and can proceed directly to the "Search" step. If the user already has information on the submission they would like to see (for example, if the user is following a link from an external site) they can go directly to that performance’s information.

### Registration For Site

If a user wishes to be a submitter, they must register for the site. If the user is going to be a reviewer (in addition to a submitter or exclusively a reviewer,) they will be given a validation code by the administrator. If no validation code is entered, the user will be registered as a submitter only.

|  |  |  |
| --- | --- | --- |
| Information Provided by User | Required | Description of Information |
| username | yes | the unique name used to log in to the site |
| password | yes | to verify their identity on repeat visits |
| first name / last name | yes | for administrative use |
| e-mail | yes | to prevent automated registrations and for administrative use |
| school affiliations / professional certifications | only for reviewer | credentials for peer-reviewers |
| validation code | only for reviewer | to verify that they are authorized to be a reviewer, not a submitter. unique code from administrator |

### Logging Into the Site

Once a user has registered, they do not need to register again. Each time they come back to the site, they will only need to log in if they are submitting something, checking on the status of a previous submission, or reviewing something.

|  |  |  |
| --- | --- | --- |
| Information Provided by User | Required | Description of Information |
| username | Yes | the unique name used to log in to the site |
| password | Yes | to verify their identity |

### Login Details Recovery

If a user forgets their password, they can have it reset, and the new one e-mailed to them.

|  |  |  |
| --- | --- | --- |
| Information Provided by User | Required | Description of Information |
| username | Yes | the username they registered with |
| e-mail address | Yes | the e-mail to send recovery information to |

### Submitting a Performance

A user must already be logged in to the web site in order to submit a performance. They must provide the information associated with the performance, as well as the actual performance piece (as an mp3 media file) to be posted automatically on the site.

|  |  |  |
| --- | --- | --- |
| Information Provided by User | Required | Description of Information |
| Performer Name(s) | yes | multiple names, one for each performer |
| Composer Name | yes | the musician who wrote the piece of music being performed |
| Title | yes | name of the piece of music being performed |
| Instruments | yes | instruments used in the performance, list all that apply |
| Year of Composition | no | the year the music was originally written ( may not be certain) |
| Year of Performance | yes | should be known to submitter |
| Live Performance | yes | yes/no |
| Time Period | yes | Chosen from a pre-defined list, such as Medieval, Renaissance, Baroque, Classical, Romantic, Modern |
| Century | no | 18th, 19th, etc. may not be known, or may be vague (composed between 1890-1910 for example) |
| File Upload | yes | the mp3 recording of the performance |

### Rating a Performance

A user must be a reviewer, and logged in, in order to rate a performance. The administrator is automatically a reviewer as well. The user will listen to the mp3 recording of the performance and then can rate the performance. The reviewer chooses whether a performance passes or doesn't pass the review, and provides comments. If there were technical problems with the file (eg. the file doesn't play for the reviewer) or there is missing information, the reviewer can choose a third option (invalid submission) which will remove the performance from the other performer's review queues (so that not every reviewer must invalidate a submission.)

|  |  |  |
| --- | --- | --- |
| Information Provided by User | Required | Description of Information |
| Invalid Submission | No | a technical problem, or information missing/incorrect |
| Passes Peer Review | yes (unless invalid is chosen) | whether the reviewer believes the performance to pass review |
| Comments | Yes | justification for review (or description of technical problem) |

### Searching for Performances

The user does not need to be logged in to search for performances. Only performances that have been accepted by all peer reviewers will be visible. Any search results that match the query will be displayed. Only limited information of the performance will be displayed.

|  |  |  |
| --- | --- | --- |
| Information from User | Required | Description of Information |
| query | yes | the keywords they are searching for |
| matching | no | what fields should be matched (title / composers / performers / etc.) If it's not provided, match all fields |

### Assigning Reviewers to Performances

The administrator needs to assign reviewers to every performance. Whenever a new performance is submitted, he will get an e-mail. Then he will assign reviewers, who will get a message that they need to review.

|  |  |  |
| --- | --- | --- |
| Information Provided by User | Required | Description of Information |
| list of reviewers | yes | the usernames of all reviewers to assign |

## Functional Requirements – Output Business Forms / Screens

### Viewing Reviews of Performances

If a user is a reviewer, they can navigate to this page and see a list of all performances they need to review, and a record of past reviews (they cannot change past reviews unless they have not been submitted yet.) It is based on their user-name so it's not required that they provide any information (but they must be logged in.)

The user will be able to see the following information

|  |  |
| --- | --- |
| Information Presented to User | Description |
| Performances | All of the performances they haven’t reviewed yet but are assigned to |
| Submitters | The person who submitted each performance |

### Reading Comments on Performance

If a user is logged in, they can view all the comments and whether their performance passed from each reviewer assigned to their performance (multiple reviewers review each performance.) They just must be logged in and the information will be provided. They don't have to provide any additional input, just navigate to the "comments" page.

|  |  |
| --- | --- |
| Information Presented to User | Description |
| Reviewers | A list of each person who did a review |
| Comments | The comments each person left |
| Passed | Whether it passed review |

### Viewing Performance Information

To view detailed information of a performance on the search results page, the user will choose which performance they want to view. No additional information will be provided, and they will be taken to a page which contains the performance information.

|  |  |
| --- | --- |
| Information Presented to User | Description |
| Performance Info | All of the detailed information of a single performance (including “listen” link) |

### Listening to a Performance

From the performance information page, a user can select a button on the page to start the playback of a performance, or a different button to download the performance.

|  |  |
| --- | --- |
| Information Presented to User | Description |
| Media Controls | Lets user play / stop / rewind the performance so they can listen to it through their speakers. |

# External Schema (User’s Views)

## Diagram of System Model of the Project

## C:\College\Fall 2009 notes\Database\SystemModel.png

## Tables before normalization

Before Normalization, our table looks something like this.

### User Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UserID | AuthorizationLevel | FirstName | LastName | Email |
| 1001 | 1 | Bob | Foobar | bf@txstate.edu |
| 1002 | 2 | Jack | Kerouac | [jk@txstate.edu](mailto:jk@txstate.edu) |
| 1003 | 2 | David | Sedaris | [ds@txstate.edu](mailto:ds@txstate.edu) |
| 1004 | 1 | Daniel | Addivare | [da@txstate.edu](mailto:da@txstate.edu) |
| 1005 | 0 | Steven | Seideman | [ss@txstate.edu](mailto:ss@txstate.edu) |

### User Table (cont’d)

|  |  |  |  |
| --- | --- | --- | --- |
| Credentials | PerformanceID | AuthorizationName | AuthorizationDescription |
| Grad School | 0001 | Reviewer | Can Review and Submit |
| Undergrad | 0002 | Submitter | Can only Submit |
| M.S. | 0003 | Submitter | Can only Submit |
| B.S. | 0004 | Reviewer | Can Review and Submit |
| PhD | NULL | Administrator | Power! |

### Performance Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PerformanceID | SubmitterName | ReviewerName | ReviewAdmissible | PassedReview |
| 0001 | Bob | Daniel | False | NULL |
| 0002 | Jack | Bob | False | NULL |
| 0003 | Jack | Daniel | True | False |
| 0004 | David | Daniel | True | True |
| 0005 | Steven | Bob | True | True |

### Performance Table (cont’d)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Performers | Composer | Title | Instruments | Century |
| June | Bach | Minuet in G | Flute,Piano | 18th |
| Delaney | Bach | Minuet in G | Piano | 18th |
| Jack, Foo | Bach | Minuet in G | Piano, Violin | 18th |
| Stelophone | Bach | Minuet in G | Viola, Violin | 18th |
| Jenny | Bach | Minuet in G | Violin | 18th |

### Performance Table (cont’d)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CompositionYear | PerformanceYear | LivePerformance | PeriodName | PeriodDescription |
| 1720 | 1995 | Yes | Baroque | Awesome Time |
| 1720 | 1999 | No | Baroque | Awesome Time |
| 1720 | 1998 | No | Baroque | Awesome Time |
| 1720 | 2007 | No | Baroque | Awesome Time |
| 1720 | 2006 | no | Baroque | Awesome Time |

## Normalized Relations

### User

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **UserID** | AuthorizationID | Password | FirstName | LastName | Email | Credentials |
| 0001 | 3 | pass | Jack | Harry | [jh@tex.com](mailto:jh@tex.com) | PhD Harvard |
| 0002 | 2 | asdf34 | Vince | Chase | [vc@entour.com](mailto:vc@entour.com) | PhD Berkeley |
| 0003 | 3 | abc123 | Jackie | Chan | [jc@howard.com](mailto:jc@howard.com) | Null |
| 0004 | 2 | cakeislie | George | Clooney | [george@clooney.com](mailto:george@clooney.com) | PhD Texas State |
| 0005 | 1 | aquarium | Admin | Istrator | [admini@gmail.com](mailto:admini@gmail.com) | PhD BrainyU |

### Submitter

|  |  |  |
| --- | --- | --- |
| **UserID** | **PerformanceID** | Date |
| 0001 | 0001 | 3/4/2009 |
| 0001 | 0002 | 3/4/2009 |
| 0001 | 0003 | 3/5/2009 |
| 0003 | 0004 | 4/5/2009 |
| 0003 | 0005 | 9/9/2009 |

### Reviewer

|  |  |  |  |
| --- | --- | --- | --- |
| **UserID** | **PerformanceID** | Date | AssignerUserID |
| 0002 | 0001 | 3/7/2009 | 0005 |
| 0002 | 0002 | 7/4/2009 | 0005 |
| 0004 | 0001 | 5/4/2009 | 0005 |
| 0005 | 0001 | 4/10/2009 | 0005 |
| 0005 | 0002 | 11/9/2009 | 0005 |

### Review

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UserID** | **PerformanceID** | Inadmissible | PassedReview | Comments |
| 0002 | 0001 | True | False | Unable to access recording |
| 0002 | 0002 | False | False | I’m sorry but your direction could use a little more work. |
| 0004 | 0001 | True | False | Couldn’t get it to load |
| 0005 | 0001 | True | False | Something was wrong with the .mp3 |
| 0005 | 0002 | False | True | Excellent performance! |

### Authorization

|  |  |  |
| --- | --- | --- |
| **AuthorizationID** | Name | Description |
| 1 | Administrator | Can assign reviewers, allow people to register as reviewers, ban unruly users, can also submit compositions. |
| 2 | Rater | Able to review submitted compositions as well as submitting compositions. |
| 3 | Submitter | Able to submit compositions and search for already-reviewed compositions. |

### PerformanceFiles

|  |  |  |
| --- | --- | --- |
| **FileID** | **PerformanceID** | URL |
| 0001 | 0001 | <http://www.fileupload.com/asdf23423> |
| 0002 | 0002 | <http://www.megaupload.com/~fileID2132> |
| 0003 | 0003 | <http://www.sharealot.com/some_file> |
| 0004 | 0004 | <http://www.ilikefiles.com/musicsubmission> |
| 0005 | 0005 | http://www.jackiechan.com/mymusic/001.mp3 |

### TimePeriod

|  |  |  |
| --- | --- | --- |
| **PeriodID** | Name | Description |
| 0001 | Renaissance | Period between 1600-1700 |
| 0002 | Baroque | 16th to early 18th century |
| 0003 | Classical | 16th to 19th century |
| 0004 | Modern | 20th century to today |
| 0005 | Middle Ages | 1300s to 1600s |

### Performance

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PerformanceID** | Performers | Composer | Title | Instruments |
| 0001 | John, Bob, Wilkins | Johnnie Vee | The Squirrel Dance | Violin, Piano |
| 0002 | Edward, Francis | Jackson Pollack | Paintings | Drums, Guitar |
| 0003 | John Williams | John Williams | Terminator | Synths, Orchestra |
| 0004 | William Tell Orchestra | Johann Sebastian Bach | The Minute Waltz | Orchestra |
| 0005 | San Francisco Acapella | Unknown | Happy Birthday, Mr. President | Vocals |

### Performance, continued

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CompositionYear | PerformanceYear | LivePerformance | PeriodID | Century |
| Null | 2009 | True | 0001 | Null |
| Null | 2008 | True | 0002 | Null |
| Null | 2007 | True | 0003 | Null |
| 1890 | 2008 | True | 0002 | 18 |
| 1925 | 2006 | True | 0003 | 19 |

## Functional Dependencies of Every Normalized Relation

### Users

AuthorizationID, Password, FirstName, LastName, Email and Credentials are functionally dependent upon UserID.

### Submitter

This keeps a record of who submits a performance, so we just have a Date. However the Date is functionally dependent on foreign keys UserID and PerformanceID so we have a complete record of who submits what.

### Performance Files

This is a reference of the location of the files for each performance. So it depends on a PerformanceID to reference which performance the file is for. Therefore, URL (the actual data) is functionally dependent on a PerformanceID and a FileID.

### Time Period

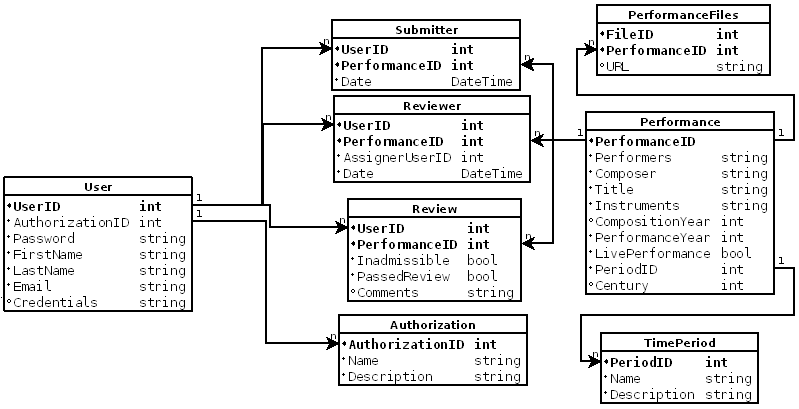
The Name and Description are information about the period, and they’re functionally dependent on the PeriodID (primary key.)

### Performance

All of the information in the table is functionally dependent on PerformanceID.

# Conceptual Schema and Logical Model of the System

## E-R Diagram of the System with Complete Tables



(preliminary)

## Physical Constraints: Number of Occurrences, Frequencies of Report, Length of Lines, Response Time for Query, Response Time for Update Transactions, Security Constraints

Currently none of these measurements exist because the project implementation has not yet begun.

# Queries and System Implementation I

## Twenty Nontrivial Queries in English with Some Complexity

### Queries the Administrator may want to do

1. Find all Registered Users
2. Remove any registered users who have not submitted or reviewed something and whose accounts have been inactive for at least 30 days
3. Find the top 5 submitters (to give them a prize!)
4. Find the percentage that each reviewer rated a submission as “invalid” vs a “pass” / “fail” rating.
5. Find 3 of the reviewers who are currently assigned to the fewest other performances (i.e. most available)

### Queries a Reviewer may want to do

1. How many reviews have I done in the last 6 months?
2. How many pieces do I have to review still?
3. Am I the top reviewer? If not, who is the top reviewer? (in # of reviews)
4. How many pieces have I rated as invalid that the other reviewers considered valid? (i.e. error rate of “invalid” marking)
5. How many pieces from each century have I reviewed?

### Queries a Submitter may want to do

1. Who else has submitted a performance of the same pieces of music I’ve submitted?
2. Am I the top submitter? If not, who is the top submitter? (in # of submissions)
3. How many performances have I submitted per month, on average?
4. How many submissions have I made that were rejected versus ones that were accepted?
5. Have my submissions been accepted more frequently in the last month than in the month prior to that (ignoring “invalid” submissions)? (am I getting better?)

### Queries a regular user may want to do

1. Find all compositions by Bach as composer
2. Find all performances that were performed live and performed after December 2007
3. Find all performances that were not performed live and were performed after December 2007
4. Are there any performances that were performed on December 12th, 2008?
5. Did anyone review the same piece of music twice by two different submitters?