Application, Logical Database Design and Architecture:

*Scorekeeper Scheduling System*

James Adams and Carri Martin

Regis University

MSSE 692 Software Engineering Practicum

Section XP40

Professor: Kevin Pyatt

May 22, 2018

**Table of Contents**

Table of Contents

[Introduction 4](#_Toc522630503)

[Purpose 5](#_Toc522630504)

[Change History/ Authors 5](#_Toc522630505)

[Application Design 6](#_Toc522630506)

[Overview of Scorekeeper Scheduling System 6](#_Toc522630507)

[Applications Design Using Wireframes by Webpage 8](#_Toc522630508)

[Login Wireframe/ Webpage 8](#_Toc522630509)

[Manage Scorekeeper/ Field Scheduling Wireframe/ Webpage 10](#_Toc522630510)

[Assign Scorekeeper to Field Wireframe/ Webpage 16](#_Toc522630511)

[Remove Scorekeeper from Field Wireframe/ Webpage 19](#_Toc522630512)

[Create Scorekeeper Wireframe/ Webpage 22](#_Toc522630513)

[Update/ Delete Scorekeeper Wireframe/ Webpage 25](#_Toc522630514)

[Application and Hardware Architecture 28](#_Toc522630515)

[Internal Application Architecture 30](#_Toc522630516)

[Logical Database Design 33](#_Toc522630517)

[Logical Entity Relationship Diagram (ERD) The following diagram is the logical ERD: 34](#_Toc522630518)

[Data Element Dictionary (DED) 35](#_Toc522630519)

# Introduction

This document details the detailed application, architecture and database design for the Scorekeeper Scheduling System. This document is structured in the following manner:

* This introduction.
* Purpose of the document.
* Document change history and authors.
* Detailed design of the application, including:
  + Application Design Overview.
  + Applications Design using Wireframes by Webpage.
  + Applications Architecture.
* Logical database design of the underlying system, including:
  + Logical Entity-Relationship Diagram (ERD).
  + Data Element Dictionary

# Purpose

This design document will detail the application, architecture and logical database design of the Scorekeeper Scheduling System and will provide a communication mechanism and the specification of design artifacts for end users, developers and stakeholders; providing the specific features and associated data and their context of how the requirements will be supported. Additionally, this document will be used to elicit feedback from the user and stakeholder community. Also, this specification will detail at a logical abstract level how the system will be logically and functionally implemented.

# Change History/ Authors

The following is the change history and authors for this document:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Reason** | **Date** | **Authors** | **E-mail** |
| V0.1 | Initial Document Creation | 5/22/2018 | James Adams and Carri Martin | [jbadams@regis.edu](mailto:jbadams@regis.edu), [cmartin007@regis.edu](mailto:cmartin007@regis.edu) |
| V0.2 | Updated for Architecture Diagram | 8/20/2018 | James Adams and Carri Martin | [jbadams@regis.edu](mailto:jbadams@regis.edu), [cmartin007@regis.edu](mailto:cmartin007@regis.edu) |

# Application Design

This section will specify the application design of the Scorekeeper Scheduling System. The first sub-section will provide an overview of the Scorekeeper Scheduling System from a user stand point. The second sub-section and its sub-sections will break out the applications design using Wireframes by Webpage.

## Overview of Scorekeeper Scheduling System

The following diagram depicts the Scorekeeper Scheduling System structure and components, which is composed of Scorekeeper Scheduling System application, including the features to manage Scorekeeper and Field schedules, including:

* Login
* Homepage – browse all field/scorekeeper schedules and set browsing date range (week, month)
* Search subset field/scorekeeper schedules by date range
* Add Scorekeeper
* Update and Delete Scorekeeper
* Add field/scorekeeper schedule entry
* Delete field/scorekeeper schedule entry

The following diagram depicts the navigation and components of the Scorekeeper Scheduling System:

Login

Manage scorekeeper/ field scheduling homepage

Create Scorekeeper

Delete/ Update Scorekeeper

Delete Scorekeeper/ Field Schedule Entry

Create Scorekeeper/ Field Schedule Entry

Features: Browse, Select Date Range and Interval

*Figure 1: Overview of the Scorekeeper Scheduling System*

## Applications Design Using Wireframes by Webpage

The following sub-sections compose the Wireframes depicting Web Pages for the Scorekeeper Scheduling System.

### Login Wireframe/ Webpage

The Login webpage allows the user to be authenticated by the Scorekeeper Scheduling System. If the user provides the correct username/userid and password they will be authenticated by the application, else they will be denied access. Also, this wireframe describes the functionality of this webpage in terms of page load events, events which are triggered when all mandatory fields are filled out and click events when push buttons or specific controls are selected:

  
*Figure 2: Login Wireframe/ Webpage*

#### Page Load Event:

The following text controls will be editable and selectable on the page load event:

* **Username**.
* **Password**.

The following is the state of the push controls and when the Login webpage loads (denoted as either enabled or disabled):

* **Login** (disabled) grayed out until all mandatory fields all populated.
* **Cancel** (enabled).

#### Mandatory Fields:

The following fields are mandatory and the **Login** push button control will be disabled until these mandatory fields are populated:

* Username
* Password

#### Click Events:

The following click events by related control and associated behaviors are associated with the click of the control:

* **Login**: when this push button control is selected, the user is validated/ authenticated against the Application\_User table, if there is a username/ userid and password match then the Manage Issue/ Resolution (homepage) is loaded, else the user gets an error invalid username/ userid and password combination, please try again.

**Note**: if the user attempts three times unsuccessfully to Login, the user is locked out and the Application\_User **inactive** column is set from False to True. There user will require unlock from the Administrator via MySQL.

* **Cancel**: when this push button control is selected, the user is prompted for confirmation and if the user confirms Yes, the user is exited from the system, if the user selects No the user is returned to the Login page.

#### Webpage Field Database Table and Column Mapping:

The following table depicts the mapping from the field on this page with the database tables and underlying columns:

| **Field**  **(Webpage)** | **Table(s)** | **Column(s)** |
| --- | --- | --- |
| Username | Application\_User | userid |
| Password | Application\_User | password |
| N/A (not visable) | Application\_User | Inactive (default True) |

### Manage Scorekeeper/ Field Scheduling Wireframe/ Webpage

The Manage Scorekeeper/ Field Scheduling Homepage - webpage provides the user with a central point to browse, search and manage (update and delete) Scorekeeper/ Field Scheduling records, and manage related support information like Scorekeepers. Also, this wireframe describes the functionality of this webpage in terms of page load events, fields contained (behavior), events which are triggered when mandatory controls are selected, fields are filled out or selected and click events when push buttons or specific controls are selected:

  
*Figure 3: Manage Scorekeeper Field Scheduling Wireframe/ Webpage*

#### Page Load Event:

The following Field/Scorekeeper Scheduling list table control will be pre-populated for the current week interval (starting on Sunday) and the fields are un-editable (read only) and will show as a standard 1 week of schedule entries in the list/table at one time (21 rows exist in the database for the date range, or less than 21 rows if one is at the end of the list, if some Fields are not scheduled every day, it will show that number of rows for that date range. Additionally, the default sort (always is by the fields date (descending default) and then by scheduled\_date (descending default):

* **Date**: this read only field is populated by the scheduled\_date column in the Field\_Scorekeeper\_Schedule table.
* **Day of Week**: this read only field is populated by a calculated value based on the scheduled\_date column in the Field\_Scorekeeper\_Schedule table.
* **Field**: this read only field is populated by the field\_name column in the Field table.
* **Game Type**: this read only field is populated by the game\_type\_description column in the Game\_Type table.
* **First Name**: this read only field is populated by the first\_name column in the Scorekeeper table.
* **Last Name**: this read only field is populated by the last\_name column in the Scorekeeper table.
* **Week Starting**: this drop-down control is populated by pre-caluated cache data for the date the current or future weeks are starting, based on every current or future Sunday.
* **Week View**: (selected/default), radio button.
* **Month View:** (not selected) , radio button.

The following is the state of the push controls, hyperlinks and radio button when the Manage Issue/ Resolution (Recipe) webpage loads (denoted as either enabled or disabled):

* **Schedule a Field Scorekeeper:** (enabled).
* **Remove Scheduled Item:** (disabled) grayed out until a Scheduled Scorekeeper/ Field List item is selected in the Scheduled Scorekeeper/ Field List (table).
* **Create Scorekeeper:** (enabled).
* **Remove Scorekeeper:** (enabled).
* **Get Date By Date Range:** (disabled) grayed out until a date is selected from Week Starting date is selected from the drop-down control.
* **Previous Date Range - hyperlink**: (enabled).
* **Next Date Range - hyperlink:** (enabled).

#### Search Custom Date Range and Related Drop-down Fields and Radio Buttons:

The Search custom date range allows the user to change the browse results for Field/Scorekeeper Scheduling list being browsed. One must select the Week Starting Date from the drop-down, and the Week View or Month View Radio button, and the select the push button control Get Date Range By Date.

Then the new set of data will be returned for the date range and interval, either weekly view of data or monthly view. Note: the default is weekly view, and the data is returned in date order.

#### Push Button Control Click Events:

The following click events by related control and associated behaviors are associated with the click of the control:

* **Schedule a Field Scorekeeper:** the click event of this push button control launches the **Assign Scorekeeper to Field** webpage which enables the user to Schedule a Scorekeeper to a Field for a specific date.
* **Remove Scheduled Item:** enabled if a Field/Scorekeeper Scheduling list item (row) is selected in the Field/Scorekeeper Scheduling list table. The click event of this push button control launches the **Remove Scorekeeper from Field** webpage which enables the user to unscheduled a scorekeeper from a field on a specific date.
* **Create Scorekeeper:** enabled by default. The click event of this push button control launches the **Create Scorekeeper** webpage which enables the user to create a Scorekeeper record in the database.
* **Remove Scorekeeper:** the click event of this push button control launches the **Remove Scorekeeper** webpage which enables the user to perform a soft delete on the scorekeeper record database, meaning that all created records remain, but the Scorekeeper cannot be added to new schedules.
* **Get Date By Date Range:** enabled if: (1) The radio button Week View (default)/ Month View is selected. (2) The date from the drop-down list Week Ending list is selected. The click event of this push button selects the related Field/Scorekeeper Scheduling list to place in the result set based on the Week or Month interval search criteria predicate which is the Week Ending date and week or month interval. When the results are returned, the date\_scheduled.

#### Webpage Field Database Table and Column Mapping:

The following table depicts the mapping from the field on this page with the database tables and underlying columns:

| **Field**  **(Webpage)** | **Table(s)** | **Column(s)** |
| --- | --- | --- |
| Date | Field\_Scorekeeper\_Schedule | scheduled\_date |
| Day of the Week | Calculated – N/A | Calculated – N/A |
| Field | Field | field\_name |
| Game Type | Game\_Type | game\_type\_description |
| First Name | Scorekeeper | first\_name |
| Last Name | Scorekeeper | last\_name |

### Assign Scorekeeper to Field Wireframe/ Webpage

The Assign Scorekeeper to Field webpage allows the user to Create a Field/ Scorekeeper Scheduling record. Also, this wireframe describes the functionality of this webpage in terms of page load events, events which are triggered when all mandatory fields are filled out and click events when push buttons or specific controls are selected:

  
*Figure 4: Assign Scorekeeper to Field Wireframe/ Webpage*

#### Page Load Event:

The following dropdown menu controls and list control will be pre-populated when the Create a Field/ Scorekeeper Scheduling webpage loads (the user still needs to select an item from the list if applicable to the Create a Field/ Scorekeeper Scheduling)

* **Select Scorekeeper**: pre-populated from the first\_name and last\_name columns in the Scorekeeper table.
* **Field**: pre-populated from the field\_name column in the Field table.
* **Game Type**: pre-populated from the game\_type\_description column in the Game\_Type table.
* **Date**: pre-populated from a cashed list of dates from the application.
* **Day of Week**: calculated when the Date is chosen by the user via the application.

The following is the state of the push controls and hyperlinks when the Create a Field/ Scorekeeper Scheduling webpage loads (denoted as either enabled or disabled):

* **Create** (disabled) grayed out until all mandatory fields all populated, which are all fields in the webpage.
* **Cancel** (enabled).

#### Mandatory Fields:

The following fields are mandatory (unless specified) and the **Create** push button control will be disabled until these mandatory fields are populated:

* First Name: read only field dependent on selection of scorekeeper from Select Scorekeeper list control.
* Last Name: read only field dependent on selection of scorekeeper from Select Scorekeeper list control.
* Field
* Game Type
* Date
* Day of Week: read only, populated when a date is chosen, and the application calculates the day of the week base on the date.

#### Click Events:

The following click events by related control and associated behaviors are associated with the click of the control:

* **Create**: when this push button control is selected, the record is created in the Field\_Scorekeeper\_Schedule table in the database. Additionally, the user is returned to the homepage when this push button control is selected.
* **Cancel**: when this push button control is selected, the user is prompted for confirmation and the record is NOT created in the Field\_Scorekeeper\_Schedule table. Additionally, the user is returned to the homepage when this push button control is selected.

#### Webpage Field Database Table and Column Mapping:

The following table depicts the mapping from the field on this page with the database tables and underlying columns:

| **Field**  **(Webpage)** | **Table(s)** | **Column(s)** |
| --- | --- | --- |
| First Name | Scorekeeper | first\_name |
| Last Name | Scorekeeper | last\_name |
| Field | Field | field\_name |
| Game Type | Game\_Type | game\_type\_description |
| Date | Field\_Scorekeeper\_Schedule | scheduled\_date |
| Day of the Week | Calculated – N/A | Calculated – N/A |
| Select Scorekeeper | Scorekeeper | first\_name, last\_name |

### Remove Scorekeeper from Field Wireframe/ Webpage

The Remove Scorekeeper from Field webpage allows the user to Delete the Field\_Scorekeeper\_Schedule record in the database. Also, this wireframe describes the functionality of this webpage in terms of page load events, and events which are triggered when the push buttons or specific controls are selected:

  
*Figure 5: Remove Scorekeeper from Field Wireframe/ Webpage*

#### Page Load Event:

The following controls will be pre-populated when the Remove Scorekeeper from Field webpage loads:

* **First Name**: is pre-populated and appears in the text box (**read only**).
* **Last Name**: is pre-populated and appears in the text box (**read only**).
* **Field**: is pre-populated and appears in the text box (**read only**).
* **Game Type**: is pre-populated and appears in the text box (**read only**).
* **Date**: is pre-populated and appears in the text box (**read only**).
* **Day of the Week**: is pre-populated, calculate base on the Date (scheduled\_date) and appears in the text box (**read only**).

The following is the state of the push controls when the Create Issue/ Resolution (Recipe) webpage loads (denoted as either enabled or disabled):

* **Removed** (enabled).
* **Cancel** (enabled).

#### Mandatory Fields:

All of the above fields in the page load event are mandatory.

#### Click Events:

The following click events by related control and associated behaviors are associated with the click of the control:

* **Remove**: when this push button control is selected, the record is deleted from the Field\_Scorekeeper\_Schedule table in the database. Additionally, the user is returned to the homepage when this push button control is selected.
* **Cancel**: when this push button control is selected, the user is prompted for confirmation and the record is NOT deleted from the Field\_Scorekeeper\_Schedule table. Additionally, the user is returned to the homepage when this push button control is selected.

#### Webpage Field Database Table and Column Mapping:

The following table depicts the mapping from the field on this page with the database tables and underlying columns:

| **Field**  **(Webpage)** | **Table(s)** | **Column(s)** |
| --- | --- | --- |
| First Name | Scorekeeper | first\_name |
| Last Name | Scorekeeper | last\_name |
| Field | Field | field\_name |
| Game Type | Game\_Type | game\_type\_description |
| Date | Field\_Scorekeeper\_Schedule | scheduled\_date |
| Day of the Week | Calculated – N/A | Calculated – N/A |

### Create Scorekeeper Wireframe/ Webpage

The Create Scorekeeper webpage allows the user to create a Scorekeeper record. Also, this wireframe describes the functionality of this webpage in terms of page load events, events which are triggered when all mandatory fields are filled out and click events when push buttons or specific controls are selected:

  
*Figure 6: Create Scorekeeper Wireframe/ Webpage*

#### Page Load Event:

The following text controls will be editable when the Create Error webpage loads:

* **First Name**.
* **Last Name**.

The following is the state of the push controls when the Create Error webpage loads (denoted as either enabled or disabled):

* **Create:** (disabled) grayed out until all mandatory fields all populated.
* **Cancel:** (enabled).

#### Mandatory Fields:

The following fields are mandatory (unless specified) and the **Create** push button control will be disabled until these mandatory fields are populated:

* First Name
* Last Name

#### Click Events:

The following click events by related control and associated behaviors are associated with the click of the control:

* **Create**: when this push button control is selected, the record is created in the Scorekeeper table in the database. Additionally, the user is returned to the **Homepage** webpage when this push button control is selected.
* **Cancel**: when this push button control is selected, the user is prompted for confirmation and the record is NOT created in the Scorekeeper table. Additionally, the user is returned to the **Homepage** webpage when this push button control is selected.

#### Webpage Field Database Table and Column Mapping:

The following table depicts the mapping from the field on this page with the database tables and underlying columns:

| **Field**  **(Webpage)** | **Table(s)** | **Column(s)** |
| --- | --- | --- |
| First Name | Scorekeeper | first\_name |
| Last Name | Scorekeeper | last\_name |

### Update/ Delete Scorekeeper Wireframe/ Webpage

The Update or Delete Scorekeeper webpage allows the user to Update or Delete a Scorekeeper record and related support information. Also, this wireframe describes the functionality of this webpage in terms of page load events, events which are triggered when all mandatory fields are filled out and click events when push buttons or specific controls are selected:

  
*Figure 7: Update/ Delete Scorekeeper Wireframe/ Webpage*

#### Page Load Event:

The following controls will be pre-populated when the Update/ Delete Error webpage loads:

* **Scorekeeper ID**: is pre-populated and appears in the text box (read only).
* **First Name**: is pre-populated and appears in the text box (editable).
* **Last Name**: is pre-populated and appears in the text box (editable).

The following is the state of the push controls when the Create Scorekeeper webpage loads (denoted as either enabled or disabled):

* **Update** (disabled) grayed out until all mandatory fields all populated and if at least one value has been changed in any of the editable fields.
* **Delete** (enabled).
* **Cancel** (enabled).

#### Mandatory Fields:

The following fields are mandatory and the **Update Error** push button control will be disabled until these mandatory fields are populated and any of the field below have changed from the values in the page load event:

* Scorekeeper ID (read only).
* First Name.
* Last Name.

#### Click Events:

The following click events by related control and associated behaviors are associated with the click of the control:

* **Update**: when this push button control is selected, the record is updated in the Scorekeeper table in the database. Additionally, the user is returned to the **Homepage** webpage when this push button control is selected.
* **Delete**: when this push button control is selected, the Scorekeeper record is soft deleted in database and the user is returned to the **Homepage** webpage.

***Note****: a soft delete marks the record so that the scorekeeper is NOT able to be added to any new Field/Scorekeeper schedule records, and it does not delete the scorekeeper from already created Field/Scorekeeper schedule records.*

* **Cancel**: when this push button control is selected, the user is prompted for confirmation and the record is NOT updated/deleted in the Scorekeeper table. Additionally, the user is returned to the **Homepage** webpage when this push button control is selected.

#### Webpage Field Database Table and Column Mapping:

The following table depicts the mapping from the field on this page with the database tables and underlying columns:

| **Field**  **(Webpage)** | **Table(s)** | **Column(s)** |
| --- | --- | --- |
| Scorekeeper ID | Scorekeeper | scorekeeper\_id |
| First Name | Scorekeeper | first\_name |
| Last Name | Scorekeeper | last\_name |

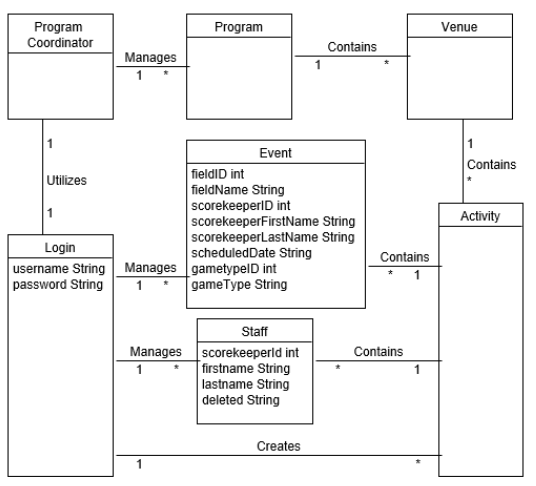
# Application and Hardware Architecture

The following sub-sections compose the Application and Hardware Architecture of the Scorekeeper Scheduling System. The envisioned hardware architecture for this project will consist of a web server to host the application and a database service to host client information in the in the Microsoft Azure Cloud. The application itself will be written in Enterprise Java utilizing a 2-tier architecture consisting of layers for Presentation, Business, Integration and Domain functionality. Within this framework, the application will employ the Model, View, Controller design pattern through the use of servlets to handle incoming requests and Java Server Pages to provide a user interface, all of which will be supported by the business, integration and domain layers. Please refer to the Figure blow:



# Internal Application Architecture

The following sub-section contains system modeling diagrams that depict the internal application architecture of the Scorekeeper Scheduling System. The Domain Layer Class Diagram below collectively models the problem domain space describing real world entities and their inter-relationships.



The Business Layer Class Diagram below depicts the interactions between the business layer managers and the service layer service classes. Each business manager is responsible for the business logic of different aspects of the system. However, they all uniformly utilize the Service Factory to call upon individual services.

Machine generated alternative text:
Business Layer Class Model 
ServiceFactory 
EvenWanager 
«lntertace» 
«lntertace» 
ListStaffDAO 
AddStaffDA01mp1 
ListStaffDAOlmpI 
«lntertace» 
Re move StaffDAO 
LoginManager 
«lnterlace» 
IService 
«lntertace» 
ScheduleEventDAO 
Stamaanager 
«lntertace» 
ListEventDAO 
«lntertace» 
LoginServiceDAO 
«lntertace» 
RemoveEventDAO 
RemoveEventDAOlmpl 
ListEventDAOlmpl 
cheduIeEventDAOlmpl 
Remove StaffDAOlmpI 
Login Service DAOImpI 

The Service Layer Class Diagram below depicts the architecture of the service layer which utilizes an interface structure to promote decoupled code and a standardized means to call upon a given service.

Service Layer Class Model 
«lnterface» 
«lntertace» 
ustStaffDAO 
AddStaffDA01mp1 
ListStaffDAOlmpl 
«lntertace» 
Remove StatfDAO 
«lntertace» 
'Service 
«lntertace» 
ScheduleEventDAO 
«lnterlace» 
ListEventDAO 
'Interface» 
LoginServiceDAO 
«lntertace» 
RemoveEventDAO 
RemoveEventDAOlmpl 
ListEventDAOlmpl 
cheduleEventDAOlmpl 
Remove StaffDAOlmpl 
Login Service DAOImpl 

# Logical Database Design

The following sub-sections compose the Logical Database design, these are:

* Logical Entity Relationship Diagram
* Data Element Dictionary (DED)

## Logical Entity Relationship Diagram (ERD) The following diagram is the logical ERD:



*Figure 8: Scorekeeper/Field Scheduling Logical ERD*

## Data Element Dictionary (DED)

The following diagram depicts the DED:

| **Table Name** | **Attribute Name** | **Description** | **Type/ Length** | **Mandatory/ Default Value** | **PK/ FK** | **FK Table Referenced** |
| --- | --- | --- | --- | --- | --- | --- |
| **Field\_ Scorekeeper\_ Schedule** | field\_id | The attribute field\_id is part of the primary key for the Field\_Scorekeeper\_Schedule table and is a foreign key to the Field table. | Int | Yes | PK/ FK | Field |
|  | scorekeeper\_id | The attribute scorekeeper\_id is part of the primary key for the Field\_Scorekeeper\_Schedule table and is a foreign key to the Scorekeeper table. | Int | Yes | PK/ FK | Scorekeeper |
|  | schedule\_date | The attribute schedule\_date is part of the primary key for the Field\_Scorekeeper\_Schedule table and is the date the scorekeeper is scheduled to a specific field. | Date | Yes | PK |  |
|  | game\_type\_id | The attribute game\_type\_id is the sequence number assign to the game type in the Game\_Type table and is a foreign key. | Int | Yes | FK | Game\_Type |
| **Field** | field\_id | The attribute field\_id is auto increment (sequence) and is the primary key for the field table. | Int | Yes | PK |  |
|  | field\_name | The Field name can be one of three names (Field 1, Field 2 or Field 3) this table is composed of 3 records currently. | Numeric (3) | Yes |  |  |
| **Scorekeeper** | scorekeeper\_id | The attribute scorekeeper\_id is auto increment (sequence) and is the primary key for the Scorekeeper table. | Int | Yes | PK |  |
|  | first\_name | The attribute first\_name is the is the scorekeepers first name. | Varchar (20) | Yes |  |  |
|  | last\_name | The attribute last\_name is the is the scorekeepers last name. | Varchar (20) | Yes |  |  |
|  | deleted | The attribute deleted, is a soft delete and is used as a predicate to filter out scorekeepers from the the list of Scorekeepers which can be selected from to assign scorekeepers to a field. This field is set from N to Y in a soft delete. | Char(1) | Yes/Default N |  |  |
| **Game\_Type** | game\_type\_id | The attribute game\_type\_id is auto increment (sequence) and is the primary key for the Game\_Type table. | Int | Yes | PK |  |
|  | game\_type\_ description | The attribute game\_type\_description is is the descriptive name of the game type being either: Mens, Mens Advanced, Coed and Coed Advanced. | Varchar (20) | Yes |  |  |