**KC Games**

# Semester Project Proposal for CSC 263 Database Systems

March 3rd, 2017

**Name of the Project Team**

Krista Forsythe

Colby Leclerc

Submitted to —

Dr. Bo Hatfield

Executive Summary 3

Problem Descriptions 4

Identifying Needs of Customers Identifying Target Specifications Potential Database Queries

Conceptual Design 7

Identify entities Identify relationships Identify constraints

Entity-Relationship Diagram

Logical Design 10

Database Schema

List of constraints that match with the schema

Project Management 14

Team Member Work Distributions Deliverables

Attachment: Scripts that create the database and establish the database constraints

KC Games aims to create a centralized web store front for customers to view and interact with game listings. Through extensive research, we've determined the most important characteristics present in current web store fronts to allow us to become a competitive business within our market vertical. We've concluded some of the most important features present in our web store front include the following: Game details; Sales information; the ability to read, write, and rate reviews; Development studios involved in development; Platforms the game supports.

Our aim is to provide a clean and concise location for users to view only the most important details of our games. Thus, we've decided on the following designs when creating our entities and their relationships. Looking at the features derived from our research, we have an entity to represent the game listing itself. This is our main entity in that we will use this game's identifier to associate all other game listing information. This entity has the main attributes associated with each game. We have Sales as its own entity, where the attributes describe the units sold and their unit price. Next, we have a review entity to store each game's zero or more reviews. Each game has one or more game studios associated with it, which we will define with the game studio entity. Lastly, each game is released on one or more platforms to play on, which we've defined as the platform entity.

Database Users

* + Front-end application - The client or customers using the store front page. They have mostly read permissions, but for reviews, they have the ability to write to the database, and change only their own reviews. Furthermore, each customer can rate a review with a +1 or –1, so this is also an update privilege. This user cannot delete any content.
  + Administrator – The user responsible for adding and updating games in the database.

Customer Needs

* + Ability to query database for a game listing. A game listing consists of the following entitles: Game, Review, Sales, Platform, Development Studio. (permission to select)
  + Ability to update review table to add review (permission to add, not delete)
  + Ability to update review table for review rating (permission to add, not delete)

Potential Queries Stories

John Smith wants to buy some games for his child’s birthday, but does not know what to get. John wants to be able to view all the top rated games for the console his child has, an Xbox 360. John then wants to see all the games in stock, and games that were released in the last 6 months. After, John wants to read a couple of reviews before making a final selection. (Colby)

Carry wants to buy a new game for her Nintendo DS, but only has enough money for one game. Carry knows what game she wants, but can’t remember the title, only the first letter of the game, and a rough estimate of a release date. If Carry saw the name of the game, she would know that’s the game she’s looking for. Carry wants to be able to search all games that start with the letter ‘S’, and ones released between February 1st 2015 and December 1st 2015. (Colby)

Jack runs a competing game-review website that launched last year, and he wants to view what other people are saying about a game before writing a review. Jack wants to search the game by the title, and sort the reviews to find the highest rated review, as well as the lowest rated review. (Colby)

Veronica just finished playing a game on her PS4 after reading an excellent review from our website. She wants to go back to the game she looked up previously, find the review by title, and +1 the review so others can see the high-quality review and possibly benefit the same way she did. (Colby)

Sarah is writing a report on a few game studios and wants to see the amount of games produced on a yearly basis. She wants to get a count of all the games released by a game studio within an interval of time. (Colby)

Michael has just finished playing a game with the genre 'Role-playing Game', and he wants to seek out other games that are in the same genre. To do this,

he can search for all games that have the genre of 'Role-playing Game' to find all games in that genre that exist in the database. (Krista)

Emily is a huge fan of games that were released in the 90's that were developed by the company Nintendo. To find games like these, she can search for all Nintendo games that were made between the years 1990 and 1999. (Krista)

Potential Queries Stories Bullets

* I would like to see other’s reviews of the game
* I would like to see the total rating for the game
* I would like to see the cover art of the game
* I would like to see the following details of the game: publisher, platform, release date, sales, price, development studio, related/suggested games.
* I would also like to see the development studio’s location
* I would like to be able to post new reviews
* I would like to rate the reviews I see with a +1 or -1
* I would like to see all games with a rating of 4 or more
* I would like to see all games with more than 1,000 sales
* I would like to see all available games made on the Nintendo 3DS
* I would like to see all games that start with the letter A
* I want to find out what category game ‘x’ falls under
* I would like to see the highest rated game
* I would like to see the highest rated review for game ‘x’
* I would like to see all games made by game studio ‘y’
* I would like to see all games that are in stock
* I would like to see all games released between January 1st 2012, and January 1st 2013
* I would like to see the newest review for game ‘x’
* I would like to see the lowest rated review for game ‘x’

Entities

* Game
  + This is our main table which we link all the other entities against. When querying for games, we will use this entity's primary key to collect the information associated with the game listing. Our project revolves around games and their characteristics in a game-listing environment (such as a store).
* Sales
  + Since our project revolves around game listings, and we're trying to imitate online store front, we need to have sale info to convey to our customers if a game is in stock, and to show the units sold, as well as the unit price to display the total sales of the game.
* Review
  + Most competitive websites have reviews for their products (Amazon being a major example) and we want to allow our customers to write, rate, and read reviews for games. A game can have any amount of reviews. Users can write their own reviews, with a star rating and a description. Furthermore, users can rate other people's reviews with a +1 or a –1.
* Platform
  + Every game has a supported platform. Some games have one, while most games have more than one platform supported. A game must have at least one platform, or else it cannot be played. A platform may be PC, Xbox, Nintendo DS, or a wide variety of other platforms. When a user sees a game listing, they want to know if they own the correct platform for the game. Thus, we want to display all supported platforms for a game.
* Development Studio
  + A game must have a creator. A development studio is a place where a group of developers create a game. A game can also have more than one development studio, where multiple studios collaborate together to create a game. When a user sees a game listing, they may want to know who created the game, since certain development studios have a better reputation for game quality than others. Thus, a customer may want to know a game's development studio before purchasing the game.

Relationships

* Game Has Sales
  + Each game has a single record defining the sales information. With the units sold and unit price data, we can derive the total revenue.
* Game Has Reviews
  + Each game has 0 or more reviews. Each review is defined as a single

record in the database, then we get all reviews associated with the foreign key Game ID primary key Review ID. Similar to sites like Amazon, a given item can have any number of reviews.

* Game Is Developed For Platforms
  + A game must be played on *something*, and we define this as it's platform. A platform is a system where the game is ran from, such as a PC, Xbox, and many more. A game must have at least 1 platform to be played on, but can be made for any number of platforms.
* Game Is Developed By Development Studio
  + A game must be created by someone or something, and in the game industry this entity is most commonly referred to as the development studio. The development studio comprises of a group of people who create the game. A game must be created by at least one studio, but can also be created by more than one studio.

Constraints

* Game Entity
  + All the attributes are NOT NULL, since each attribute must be present for a game listing to be valid, and will not be NULL at any point in time.
  + A game as a primary key GameID which we will use to uniquly identify each record, and use this key to relate the record to all other entities.
* Sales Entity
  + All the attributes are NOT NULL, but default to 0. A game can either have no sales, or more than 1 sale, thus NULL will never be present.
  + Using a primary key SalesID matched with the foreign key GameID, a Game record can only have one Sales record.
* Review Entity
  + All the attributes are NOT NULL, since each attribute must be present with valid data before being accepted as a review. At no point in time are any attributes NULL.
  + Using a linking table with a primary key of ReviewID and a foreign key GameID, each review belongs to only one Game, but many reviews can belong to one Game.
* Development Studio Entity
  + All attributes of Development Studio are NOT NULL, since a studio must have an address and a name, meaning these attributes will never be NULL.
* Platform Entity
  + All attributes of Platform are NOT NULL, since each platform must have a name, company who created it, and a release date. At no point in time are

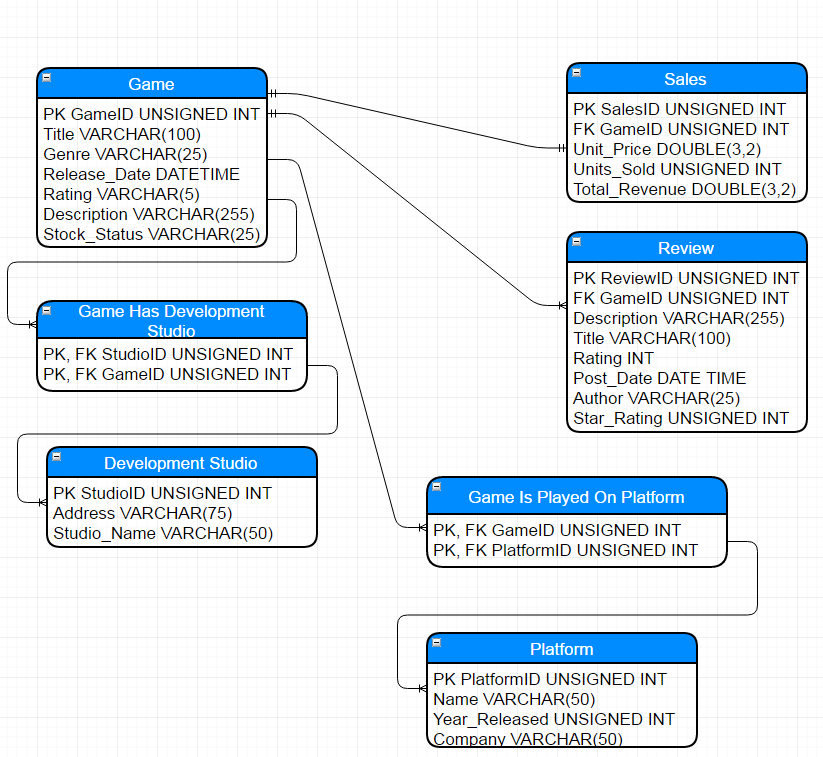
any of these attributes NULL.

* + Using a linking table, each platform is defined with a primary key PlatformID, and associated with a foreign key GameID.

Business Rules

* Game has sales
* Game has one sale, and sale can have one game
* Attributes of game are game id, title, genre, suggested, release date, rating, description, image, stock status
* Attributes of sale are sales id, unit price, unit sold, total price
* A game has many reviews, reviews has one game
* Review has attributes review id, description, title, rating, post date, author, and star rating
* Game has many or one development studio
* Development studio makes many games
* Development studio has attributes address and studio name
* Game is made on one or more platforms
* A platform has one or many games
* Platform has attributes platform id, name, year released, and company

/Users/colbyschool/Documents/School/Salem State/CSC 263/Project/ERD Diagrams/KCGamesERD.png



Final List of Constraints:

* Game: Primary Key (GameID), NOT NULL for all attributes.
* Sales: Primary Key (SalesID), Foreign Key (GameID), NOT NULL for all attributes.
* Review: Primary Key (ReviewID), Foreign Key (GameID), NOT NULL for all attributes.
* Game Is Played On Platform: Primary Key and Foreign Key (PlatformID, GameID), NOT NULL for all attributes.
* Platform: Primary Key (PlatformID), NOT NULL for all attributes.
* Game Has Development Studio: Primary Key and Foreign Key (StudioID, GameID), NOT NULL for all attributes.
* Development Studio: Primary Key (StudioID), NOT NULL for all attr

DROP TABLE IF EXISTS Game, Sales, Review, Platform, G\_CreatedAt\_DS, G\_PlayedOn\_P, Development\_Studio;

CREATE TABLE Game (

GameID INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY,

Title VARCHAR(100) NOT NULL,

Genre VARCHAR(25) NOT NULL,

Release\_Date DATETIME NOT NULL,

Rating VARCHAR(5) NOT NULL,

Description VARCHAR(255) NOT NULL,

Stock\_Status VARCHAR(25) NOT NULL

);

CREATE TABLE Sales (

SalesID INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY,

GameID INTEGER NOT NULL,

Unit\_Price DOUBLE NOT NULL,

Units\_Sold INTEGER NOT NULL,

Total\_Revenue DOUBLE NOT NULL,

FOREIGN KEY (GameID) REFERENCES Game(GameID)

);

CREATE TABLE Review (

ReviewID INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY,

GameID INTEGER NOT NULL,

Description VARCHAR(255) NOT NULL,

Title VARCHAR(100) NOT NULL,

Rating INTEGER NOT NULL,

Post\_Date DATETIME NOT NULL,

Author VARCHAR(25) NOT NULL,

Star\_Rating INTEGER NOT NULL,

FOREIGN KEY (GameID) REFERENCES Game(GameID)

);

CREATE TABLE Platform (

PlatformID INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY,

Name VARCHAR(50) NOT NULL,

Year\_Released INTEGER NOT NULL,

Company VARCHAR(50) NOT NULL

);

CREATE TABLE G\_PlayedOn\_P (

PlatformID INTEGER NOT NULL,

GameID INTEGER NOT NULL,

FOREIGN KEY (GameID) REFERENCES Game(GameID),

FOREIGN KEY (PlatformID) REFERENCES Platform(PlatformID),

PRIMARY KEY (GameID, PlatformID)

);

CREATE TABLE Development\_Studio (

StudioID INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY,

Address VARCHAR(75) NOT NULL,

Studio\_Name VARCHAR(50) NOT NULL

);

CREATE TABLE G\_CreatedAt\_DS (

StudioID INTEGER NOT NULL,

GameID INTEGER NOT NULL,

FOREIGN KEY (GameID) REFERENCES Game(GameID),

FOREIGN KEY (StudioID) REFERENCES Development\_Studio(StudioID),

PRIMARY KEY (GameID, StudioID)

);

## Project Management

Colby Leclerc

Colby Leclerc is a student working towards his Bachelor of Science in Computer Science. Colby is a Junior Software Engineer, and has experience with Java, Python, C, Scala, REST and SOAP API’s, JavaScript, SQL (MSSQL 2008, MySQL, and PostgreSQL), HTML and CSS. Colby has been involved with programming and computer science since he was 14, and has developed multiple open sourced game plugins for game server communities. His responsibilities include finding data for the Game table, Sales table, Review table, and Platform table.

Krista Forsythe

Krista Forsythe is a student working towards her Bachelor of Science in Computer Science. Krista is an IT professional with experience in PHP, Python, Java, SQL, CSS, and HTML. She has been learning website development and database querying since Middle School. Her responsibilities include finding data for the Game table, Sales table, Development Studio table, and Platform table.