Game Studio Database

Student Name: Richard Whitney

Student Number: 20040645

Date: 04/11/18

Module: Database

Table of Contents

Business Description

Game development is the process of creating video games. This is preformed by a game studio, which can range from a single person to a large team spread across the world. For the purpose of this project, I will be modelling a mid-sized studio with multiple people working in different departments. Traditional PC and console games can take several years to reach completion. The development of games is usually funded by publishers.

Most studios work on several games at once. This allows studios to receive a steady stream of revenue as games are released at different times. Generally, studios increase the number of employees working on a title as it enters full production and reallocate or scale back resources upon completion.

Most games are developed in phases. In pre-production game design documents are written, prototypes are developed, and pitches are put forward to financiers. If an idea is approved, full-scale development starts. This will involve a team of people with various roles in different departments, including designers, artists, programmers, and testers.

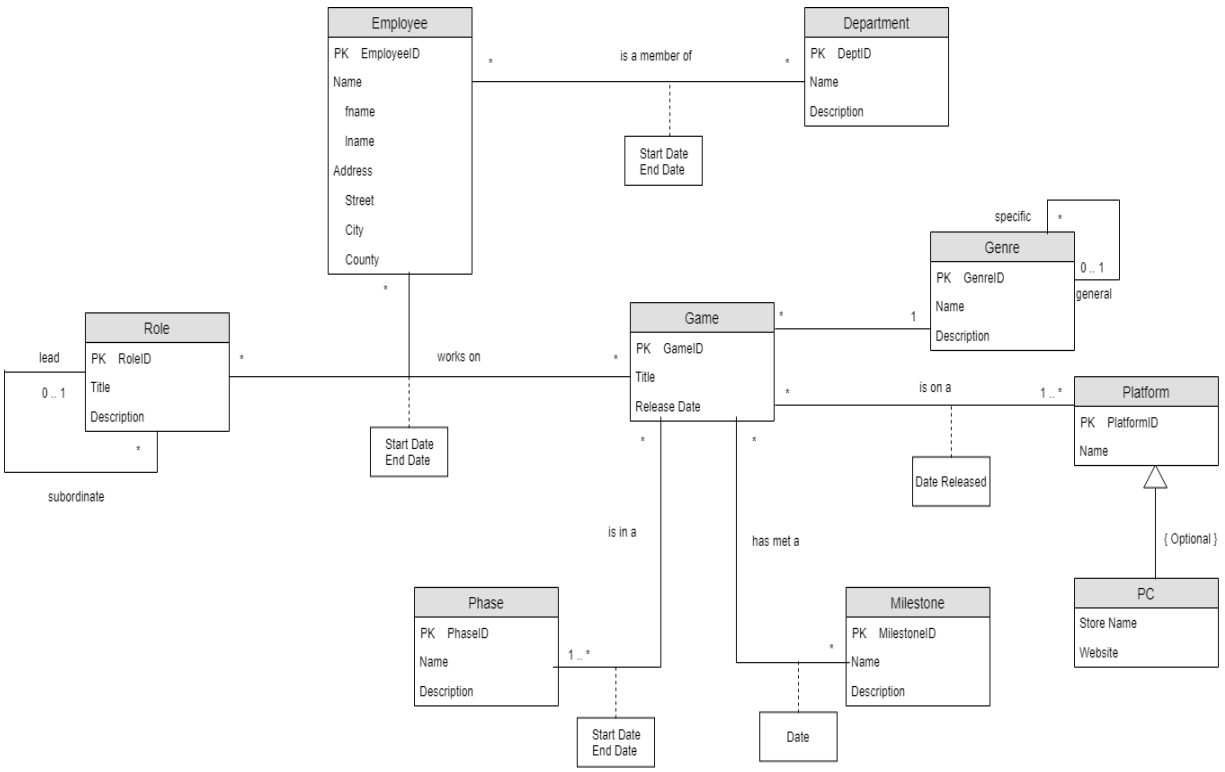
Much like other entertainment media, games can be categorised by one or several genres. Designers often experiment with different combinations of genres. Games are released on different platforms. These platforms are traditional current generation consoles an PC. The length of development is influenced by a number of factors, including, scale, genre and development platform.

Games may be required to meet milestones during development set by publishers. Milestones mark major points during a game's development and are used to track its progress. Example milestones may include: first playable, alpha, or beta, game versions.

The implementation of a database will allow the studio to do as follows:

- Keep track of which employee is working or has worked each game. This could be useful in determining performance metrics and assigning bonuses based on a game's success.
- Manage employees more efficiently. The studio will be able to see how many employees are working on each game at each stage of its development. The studio could reassign people to different games base on each games stage of development.
- Determine what games are currently in development and each phase they are in.
- Determine what milestone are game has achieved and what the next one is. Publishers usually release funds to the studio when each milestone is reached.
- Perform an analysis on past development. For example, determine which genre takes the longest to develop, which milestone takes the longest to reach. This would enable the studio to more accurately predict the development times of future games.

Enhanced ER Diagram



Normalised Tables

Employee (employeeID, fName, lName, street, city, county)

Primary key employeeID

Department (deptID, name, description)

Primary key deptID

Role (roleID, title, description, parentID)

Primary key roleID

Foreign key parentID references Role(roleID)

Game (gameID, title, releaseDate, genreID)

Primary key gameID

Foreign key genreID references Genre(genreID)

Genre (genreID, name, description, parentID)

Primary key genreID

Foreign key parentID references Genre(genreID)

Phase (phaseID, name, description)

Primary key phaseID

Milestone (milestoneID, name, description)

Primary key milestoneID

Platform (platformID, name)

Primary key platformID

PC (platformID, storeName, website)

Primary key platformID

Foreign key platformID references Platform(platformID)

IsAMemberOf (employeeID, deptID, startDate, endDate)

Primary key employeeID, deptID, startDate

Foreign key employeeID references Employee(employeeID)

Foreign key deptID references Department(deptID)

WorksOn (employeeID, gameID, roleID, startDate, endDate)

Primary key employeeID, gameID, roleID

Foreign key employeeID references Employee(employeeID)

Foreign key gameID references Game(gameID)

Foreign key roleID references Role(roleID)

IsInA (gameID, phaseID, startDate, endDate)

Primary key gameID, phaseID

Foreign key gameID references Game(gameID)

Foreign key phaseID references Phase(phaseID)

HasMetA (gameID, milestoneID, date)

Primary key gameID, milestoneID

Foreign key gameID references Game(gameID)

Foreign key milestoneID references Milestone(milestoneID)

IsOnA (gameID, platformID, dateReleased)

Primary key gameID, platformID

Foreign key gameID references Game(gameID)

Foreign key platformID references Platform(platformID)

Table Design

Employee

Stores information for each employee, such as name and address. Each employee will be part of a department and can hold many roles during development of a game at any time.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
EmployeeID	Integer		Not Null		Unique	PK	Unique employee number
FName	Char	20	Not Null				First name of employee
LName	Char	20	Not Null				Last name of employee
Street	Char	30					Street name, part of address
City	Char	30					City name, part of address
County	Char	30					County name, part of address

Department

Stores information for each department, such as name and description. Each department will have one or more employees. The department entity is the primary means for grouping employees by skill sets.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
DeptID	Integer		Not Null		Unique	PK	Unique department number
Name	Char	20	Not Null				Name of the department
Description	Varchar	256					Description of the department

Role

Stores information for each role that an employee has when working on a game, such as title and hierarchical organisation. Lead and subordinate roles are separate entities. An employee can have one or more roles on a game. Each role can lead one or more other roles. The role entity is the primary means for a hierarchical representation of the employees for any game. The role entity forms a relationship with the employee and game entity to record the employee’s work history.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
RoleID	Integer		Not Null		Unique	PK	Unique role number
Title	Char	20	Not Null				The name of the role
Description	Varchar	256					The description of the role
ParentID	Integer					FK	The roleID of the parent role

Game

Stores information on each game that the studio is working on, or has released. Each game will have multiple employees filling one or more roles during different phases of development. Milestones will be recorded as they are met during a game's development. Each game belongs to one genre, and is released on one or more platforms.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
GameID	Integer		Not Null		Unique	PK	Unique game number
Title	Varchar	20	Not Null				The name of the game
Release Date	Date						The planned release date of the game
GenreID	Integer		Not Null			FK	The ID of the genre the game belongs to

Genre

Stores information about the genre of a game, such as title and description. Each game will belong to exactly one genre. The genre table has a recursive relationship with itself to represent possible sub-genres that exist.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
GenreID	Integer		Not Null		Unique	PK	Unique genre number
Name	Varchar	32	Not Null		Unique		Name of genre
Description	Varchar	256					Description of genre
ParentID	Integer					FK	The ID of the parent genre

Phase

Stores information on a phase that a game can be in during its development, such as the name of the phase and a description. Each game will be in exactly one phase at any given point of its development.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
PhaseID	Integer		Not Null		Unique	PK	Unique phase number
Name	Varchar	20	Not Null		Unique		Name of the phase
Description	Varchar	256					Description of the phase

Milestone

Stores information about a milestone, such as name. Each game will meet one or more milestones during its development. Each milestone is met at most once, at which point it is given a date stamp.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
MilestoneID	Integer		Not Null		Unique	PK	Unique milestone number
Name	Varchar	32	Not Null		Unique		Name of the milestone
Description	Varchar	256					Description of the milestone

Platform

Stores information about a platform, such as its name. Each game will be release on one or more platforms. This entity has one optional sub-type that stores additional information.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
PlatformID	Integer		Not Null		Unique	PK	Unique platform number
Name	Varchar	32	Not Null		Unique		Name of the platform

PC

This entity is a sub-type of the platform entity. It stores additional information about games released for PC. The PC platform has its own platforms (hence referred to stores). PC games will be released on one or more stores.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
PlatformID	Integer		Not Null		Unique	PK	Unique platform number
Store Name	Varchar	32	Not Null		Unique		Name of the store
Website	Varchar	50					The web address of the store

Is a Member of

This entity records the relationship between employees and departments. Each employee can be a member of only one department at any given time, but may be a member of different departments during their time at the studio. A department can have many employees at any given time. The start date and end date of the employee's time at the department is recorded. The combination of employeeID, deptID and startDate is unique and is the primary key.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
EmployeeID	Integer		Not Null			Part of PK FK	The ID of the employee
DeptID	Integer		Not Null			Part of PK FK	The ID of the department
Start Date	Date		Not Null	Today		Part of PK	The date the employee started working at the department
End Date	Date				Must be null or >= Start Date		The date the employee left the department

Works On

This entity records the ternary relationship between employee, role and game. Each employee will work on many games, at different roles during their time at the studio. The start date and end date of an employee's time working on a game, for a particular role is recorded. The combination of employeeID, gameID and roleID is unique and is the primary key

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
EmployeeID	Integer		Not Null			Part of PK FK	The ID of the employee
GameID	Integer		Not Null			Part of PK FK	The ID of the game
RoleID	Integer		Not Null			Part of PK FK	The ID of the role
Start Date	Date		Not Null	Today			The date an employee started working on a game, for a given role
End Date	Date				Must be null or >= Start Date		The date an employee stopped working on a game, for a given role

Is In a

This entity records the relationship between a game and a phase. Each game must be in exactly one phase at any given time, but may be in multiple phases during the course of its development. The start date and end date of when a game enter and exited a phase is recorded. The combination of gameID and phaseID is unique and is the primary key.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
GameID	Integer		Not Null			Part of PK FK	The ID of the game
PhaseID	Integer		Not Null			Part of PK FK	The ID of the phase
Start Date	Date		Not Null	Today			The date that a game entered a particular phase
End Date	Date				Must be null or >= Start Date		The date that a game finished a particular phase

Has Met a

This entity records the relationship between a game and a milestone. Each game will normally met one or more milestones during its development. The date a milestone was met is recorded. The combination of gameID and milestoneID is unique and is the primary key.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
GameID	Integer		Not Null			Part of PK FK	The ID of the game
MilestoneID	Integer		Not Null			Part of PK FK	The ID of the milestone
Date	Date		Not Null	Today			The date that the milestone was met

Is On a

This entity records the relationship between a game and a platform. Each game will release on at least one platform. A platform will have many game releases. The date a game was released on a platform is recorded. The combination of gameID and platformID is unique and is the primary key.

Field	Type	Size	Null/Not Null	Default	Constraints	Index	Description
GameID	Integer		Not Null			Part of PK FK	The ID of the game
PlatformID	Integer		Not Null			Part of PK FK	The ID of the platform
Date Released	Date		Not Null	Today			The date a game released on the platform