University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: 27 September, 2024

Group Number: 44

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Kshitij Gomber	18521526	n4i6m	kshitijgomber@gmail.com
Luke Nathan	38946877	a2x9g	lnathan80@gmail.com
Apoorva Devarakonda	66647223	z1o3g	devarka403@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

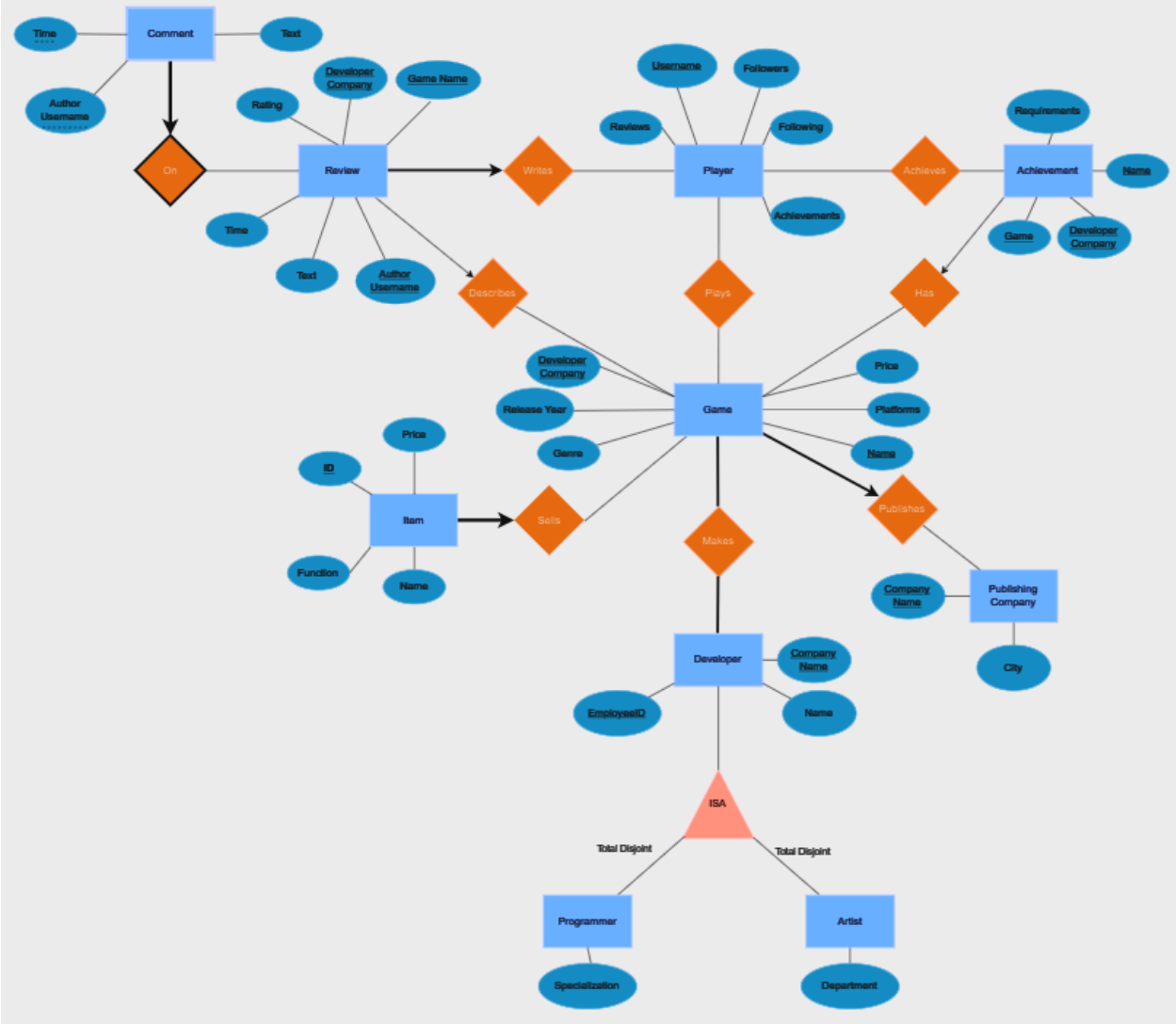
In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

University of British Columbia, Vancouver
Department of Computer Science

Project Summary

Our project is a database representation of a website that allows users to log and review games, track their development, follow other users, and track in-game item purchases. Games are the central focus of the application, with all interactions, including those between users, centered around them.

ER Diagram



The diagram has changed from milestone 1 to accommodate for foreign keys. Both the Achievement and Review entities now have a “Developer Company” key.

University of British Columbia, Vancouver
Department of Computer Science

Schema

1.Player(username: VARCHAR(50), followers: INT, following: INT, reviews: INT , achievements: INT)
Primary Key: username
Constraints: username NOT NULL, UNIQUE(username)

2.Achievement(requirements: VARCHAR(255), name: VARCHAR(100), game_name: VARCHAR(100), developing_company:VARCHAR(100))
- Primary Key: name, developing_company, game_name
-Foreign Key: developing_company, game_name
- Constraints: name NOT NULL, UNIQUE(name), developing_company NOT NULL, game_name NOT NULL

3.Comment(time: TIME, author_username: VARCHAR(50), text: TEXT ,game_name: VARCHAR(100), developing_company:VARCHAR(100))
- Primary Key: time, author_username, game_name, developing_company
- Foreign Keys: author_username, game_name,developing_company
- Constraints: time NOT NULL, author_username NOT NULL, game_name NOT NULL,review_author_username NOT NULL,developing_company NOT NULL.

4. Review(rating: INT, game_name: VARCHAR(100), developing_company: VARCHAR(100), time: TIME, text: TEXT, author_username: VARCHAR(50))
- Primary Key: game_name, author_username
- Candidate Key: author_username, time
- Foreign Key: author_username, game_name,developing_company
- Constraints: game_name NOT NULL, author_username NOT NULL, game_developing_company NOT NULL,.

5. Item(id: INT, price: DECIMAL(10, 2), function: VARCHAR(255), name: VARCHAR(100), game_name: VARCHAR(100), developing_company: VARCHAR(100))
- Primary Key: id
-Candidate Key: function
- Foreign Key: game_name, developing_company)
- Constraints: id NOT NULL, UNIQUE(id), price NOT NULL, function NOT NULL,UNIQUE(function), name NOT NULL, game_name NOT NULL, developing_company NOT NULL

6.Publishing_Company(company_name: VARCHAR(100), city: VARCHAR(100))
- Primary Key: company_name
- Constraints: company_name NOT NULL, city NOT NULL

7. Programmer(company_name: VARCHAR(100), employee_id: INT, name: VARCHAR(100), specialization: VARCHAR(100))

- Primary Key: company_name, employee_id.
- Constraints: company_name NOT NULL, employee_id NOT NULL, employee_id(UNIQUE), name NOT NULL, specialization NOT NULL

8. Artist(company_name: VARCHAR(100), employee_id: INT, name: VARCHAR(100), department: VARCHAR(100))
- Primary Key: company_name, employee_id.
- Constraints: company_name NOT NULL, employee_id NOT NULL,employee_id(UNIQUE) name NOT NULL, department NOT NULL,

9. Game(name: VARCHAR(100), developing_company: VARCHAR(100), release_year: INT, genre: VARCHAR(50), price: DECIMAL(10, 2), platform: VARCHAR(100))
- Primary Key: name, developing_company
- Foreign Key: publishing_company
- Constraints: name NOT NULL, developing_company NOT NULL, release_year NOT NULL, genre NOT NULL, price NOT NULL, platform NOT NULL, publishing_company NOT NULL

Functional Dependencies

Game:
Developer Company, Name, -> Price, Release Year, Platforms, Genre
Name, Genre, Release Year , Platforms-> Developer Company

Developer:
Company Name, Employee ID -> Name

Publishing Company
Company Name -> City

Item
ID -> Name, Function, Price
Name -> Function
Name, Function -> Price

Player
Username -> Following, Followers, Reviews, Achievements

Achievement
Name, Game, Developer Company -> Requirements
Developer Company, Game, Requirements -> Name

Review
Developer Company, Game Name, Author Username -> Text, Time, Rating

University of British Columbia, Vancouver
Department of Computer Science

Time, Author Username -> Developer Company, Game Name, Text, Rating

Comment
Time, Author Username, Review Game Name, Review Developer Company, Review Author Username -> Text
Time, Author Username, Review Time, Review Author Username -> Text

Normalization

1. Game Relation

Attributes:

(Developer Company, Name, Price, Release Year, Platforms, Genre)

Functional Dependencies (FDs):

Developer Company, Name → Price, Release Year, Platforms, Genre

Name, Genre, Release Year, Platforms → Developer Company

Step 1: Check if Game is in BCNF

In FD 1, Developer Company, Name is a superkey (since it determines all other attributes).

In FD 2, Name, Genre, Release Year, Platforms is not a superkey. This violates BCNF.

Step 2: Decompose on FD 2:

R1: (Name, Genre, Release Year, Platforms, Developer Company)

R2: (Developer Company, Name, Price, Release Year, Platforms, Genre)

Step 3: Check R1 and R2

R1 is in BCNF (two FDs but both involve the full set or subsets of attributes).

R2 now has a single FD: Developer Company, Name → Price, Release Year, Platforms, Genre. This relation is in BCNF.

BCNF Decomposed Relations for Game:

R1(Name, Genre, Release Year, Platforms, Developer Company)

R2(Developer Company, Name, Price, Release Year, Platforms, Genre)

2. Developer Relation

Attributes:

(Company Name, Employee ID, Name)

Functional Dependencies:

Company Name, Employee ID → Name

Check if Developer is in BCNF:

Company Name, Employee ID is a superkey, so the relation is in BCNF.

BCNF Relation for Developer:

Developer(Company Name, Employee ID, Name)

3. Publishing Company Relation

Attributes:

(Company Name, City)

Functional Dependencies:

Company Name → City

Check if Publishing Company is in BCNF:

Company Name is a superkey, so the relation is in BCNF.

BCNF Relation for Publishing Company:

Publishing_Company(Company Name, City)

4. Item Relation

Attributes:

(ID, Name, Function, Price)

Functional Dependencies:

ID → Name, Function, Price

Name → Function

Name, Function → Price

Step 1: Check if Item is in BCNF

FD 1: ID is a superkey, so this does not violate BCNF.

FD 2: Name is not a superkey, so this violates BCNF.

Step 2: Decompose on FD 2:

R1: (Name, Function)

R2: (ID, Name, Price)

Step 3: Check R1 and R2

R1(Name, Function) is in BCNF.

R2(ID, Name, Price) still has ID → Name, Price. Since ID is a superkey, it is in BCNF.

BCNF Decomposed Relations for Item:

R1(Name, Function)

R2(ID, Name, Price)

5. Player Relation

Attributes:

(Username, Following, Followers, Reviews, Achievements)

Functional Dependencies:

Username → Following, Followers, Reviews, Achievements

Check if Player is in BCNF:

Username is a superkey, so the relation is in BCNF.

BCNF Relation for Player:

Player(Username, Following, Followers, Reviews, Achievements)

6. Achievement Relation

Attributes:

(Name, Game, Developer Company, Requirements)

Functional Dependencies:

Name, Game Developer Company → Requirements

Game, Developer Company, Requirements → Name

Check if Achievement is in BCNF

Both Name, Game, Developer Company and Game, Developer Company, Requirements are superkeys. So, this relation is already in BCNF.

BCNF Relation for Achievement:

Achievement(Name, Game, Requirements)

7. Review Relation

Attributes:

(Game Name, Author Username, Text, Time, Rating)

Functional Dependencies:

Game Name, Developer Company, Author Username → Text, Time, Rating

Time, Author Username → Game Name, Developer Company, Text, Rating

Step 1: Check if Review is in BCNF

FD 1: Game Name, Developer Company, Author Username is a superkey, so this is fine.

FD 2: Time, Author Username is not a superkey, so this violates BCNF.

Step 2: Decompose on FD 2:

R1: (Time, Author Username, Game Name, Developer Company, Text, Rating)

R2: (Time, Author Username)

BCNF Decomposed Relations for Review:

R1(Time, Author Username, Game Name, Developer Company, Text, Rating)

R2(Time, Author Username)

8. Comment Relation

Attributes:

(Time, Author Username, Review Game Name, Review Author Username, Text)

Functional Dependencies:

Time, Author Username, Review Game Name, Review Developer Company, Review Author Username → Text

Time, Author Username, Review Time, Review Author Username → Text

Check if Comment is in BCNF:

Both FDs have their LHS as superkeys, so this relation is already in BCNF.

BCNF Relation for Comment:

Comment(Time, Author Username, Review Game Name, Review Author Username, Text)

Final Decomposed BCNF Relations

Game:

R1(Name, Genre, Release Year, Platforms, Developer Company)

R2(**Developer Company**, **Name**, Price, Release Year, Platforms, Genre)

Developer:

University of British Columbia, Vancouver
Department of Computer Science

Developer(**Company Name**, **Employee ID**, Name)

Publishing Company:

Publishing_Company(Company Name, City)

Item:

R1(Name, Function)

R2(ID, Name, Price)

Player:

Player(Username, Following, Followers, Reviews, Achievements)

Achievement:

Achievement(Name, Game, Requirements)

Review:

R1(Time, **Author Username**, **Game Name**, **Developer Company**, Text, Rating)

R2(Time, Author Username)

Comment:

Comment(Time, *Author Username*, Review Game Name, Review Author Username, Text)

SQL DDL

1.

```
CREATE TABLE Player (
   username VARCHAR(50) NOT NULL,
   followers INTEGER,
   following INTEGER,
   reviews INTEGER,
   achievements INTEGER,
   PRIMARY KEY (username),
```

```
   UNIQUE (username)
);
```

Note: we cannot currently show many to many relationships of Player entity. Will be enforcing them later.

2.
```
CREATE TABLE Achievement (
   name VARCHAR(100) NOT NULL,
   game_name VARCHAR(100) NOT NULL,
   developer_company VARCHAR(100) NOT NULL,
   requirements VARCHAR(255) ,
   PRIMARY KEY (name, game_name),
   UNIQUE (name, game_name),
   FOREIGN KEY (game_name) REFERENCES Game(name) ON DELETE CASCADE
);
```

3.
```
CREATE TABLE Comment (
   time TIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
   author_username VARCHAR(50) NOT NULL,
   text TEXT ,
   game_name VARCHAR(100) NOT NULL,
   game_developing_company VARCHAR(100) NOT NULL,
   PRIMARY KEY (time, author_username, review_game_name,
review_game_developing_company),
   FOREIGN KEY (author_username) REFERENCES Player(username) ON DELETE CASCADE,
   FOREIGN KEY (game_name, game_developing_company)
      REFERENCES Review(game_name, game_developing_company) ON DELETE CASCADE
);
```

4.
```
CREATE TABLE Review (
   rating INTEGER NOT NULL,
   game_name VARCHAR(100) NOT NULL,
   game_developing_company VARCHAR(100) NOT NULL,
   time TIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
   text TEXT NOT NULL,
   author_username VARCHAR(50) NOT NULL,
   PRIMARY KEY (game_name, author_username),
   FOREIGN KEY (author_username) REFERENCES Player(username) ON DELETE CASCADE,
   FOREIGN KEY (game_name, game_developing_company) REFERENCES Game(name,
developing_company) ON DELETE CASCADE,
 );
```

```
5. CREATE TABLE Item (
   id INT NOT NULL,
   price DECIMAL(10, 2) NOT NULL,
   function VARCHAR(255) NOT NULL,
   name VARCHAR(100) NOT NULL,
   game_name VARCHAR(100) NOT NULL,
   developing_company VARCHAR(100) NOT NULL,
   PRIMARY KEY (id),
   UNIQUE (id),
   UNIQUE (function),
   FOREIGN KEY (game_name, developing_company) REFERENCES Game(name,
developing_company) ON DELETE CASCADE
);

6.CREATE TABLE Publishing_Company (
   company_name VARCHAR(100) NOT NULL,
   city VARCHAR(100) NOT NULL,
   PRIMARY KEY (company_name)
);

7.CREATE TABLE Programmer (
   company_name VARCHAR(100) NOT NULL,
   employee_id INT NOT NULL,
   name VARCHAR(100) NOT NULL,
   specialization VARCHAR(100) NOT NULL,
   PRIMARY KEY (company_name, employee_id),
   UNIQUE (employee_id),
  );

8.  CREATE TABLE Artist (
   company_name VARCHAR(100) NOT NULL,
   employee_id INTEGER NOT NULL,
   name VARCHAR(100) NOT NULL,
   department VARCHAR(100) NOT NULL,
   PRIMARY KEY (company_name, employee_id),
   UNIQUE (employee_id),
 );

9. CREATE TABLE Game (
   name VARCHAR(100) NOT NULL,
   developing_company VARCHAR(100) NOT NULL,
   release_year INTEGER NOT NULL,
   genre VARCHAR(50) NOT NULL,
```

```
    price DECIMAL(10, 2) NOT NULL,
    platform VARCHAR(100) NOT NULL,
    publishing_company VARCHAR(100) NOT NULL,
    PRIMARY KEY (name, developing_company),
    FOREIGN KEY (publishing_company) REFERENCES Publishing_Company(company_name) ON
DELETE CASCADE
);
```

INSERT Statements

1. Player Table

```
INSERT INTO Player (username, followers, following, reviews, achievements) VALUES

('kshitij', 50, 10, 4, 2),

('lukeN', 35, 8, 5, 1),

('apoorvaD', 42, 9, 3, 3),

('gamingLegend', 100, 50, 12, 7),

('casualGamer', 20, 3, 1, 0);
```

2. Achievement Table

```
INSERT INTO Achievement (name, game_name, developer_company, requirements) VALUES

('First Blood', 'Battle Royale', 'Epic Games', 'Get first kill in match'),

('Speedrunner', 'Fast Run', 'Speed Studios', 'Complete game under 2 hours'),

('Collector', 'Adventure Quest', 'Mystic Games', 'Collect all items'),

('Veteran', 'War Zone', 'Epic Games', 'Play 100 matches'),

('Puzzle Master', 'Mystery Island', 'Mystic Games', 'Solve all puzzles');
```

3. Comment Table

INSERT INTO Comment (time, author_username, text, game_name, game_developing_company) VALUES

('12:15:30', 'kshitij', 'This game is amazing!', 'Battle Royale', 'Epic Games'),

('14:10:00', 'lukeN', 'Loved the mechanics!', 'Fast Run', 'Speed Studios'),

('16:45:00', 'apoorvaD', 'Could use some improvements.', 'Adventure Quest', 'Mystic Games'),

('18:30:25', 'gamingLegend', 'Best multiplayer experience.', 'War Zone', 'Epic Games'),

('21:15:00', 'casualGamer', 'Puzzles were too hard.', 'Mystery Island', 'Mystic Games');

4. Review Table

INSERT INTO Review (rating, game_name, game_developing_company, time, text, author_username) VALUES

(5, 'Battle Royale', 'Epic Games', '12:30:00', 'Outstanding gameplay!', 'kshitij'),

(4, 'Fast Run', 'Speed Studios', '14:20:00', 'Really fast-paced and fun.', 'lukeN'),

(3, 'Adventure Quest', 'Mystic Games', '17:00:00', 'Average but fun.', 'apoorvaD'),

(5, 'War Zone', 'Epic Games', '19:00:00', 'Fantastic multiplayer!', 'gamingLegend'),

(2, 'Mystery Island', 'Mystic Games', '21:30:00', 'Too difficult for me.', 'casualGamer');

5. Item Table

INSERT INTO Item (id, price, function, name, game_name, developing_company) VALUES

(1, 9.99, 'Health boost', 'Med Kit', 'Battle Royale', 'Epic Games'),

(2, 4.99, 'Speed boost', 'Turbo Boots', 'Fast Run', 'Speed Studios'),

(3, 14.99, 'Unlock levels', 'Golden Key', 'Adventure Quest', 'Mystic Games'),

(4, 2.99, 'Extra life', 'Revive Token', 'War Zone', 'Epic Games'),

(5, 19.99, 'Hint reveal', 'Puzzle Solver', 'Mystery Island', 'Mystic Games');

6. Publishing_Company Table

INSERT INTO Publishing_Company (company_name, city) VALUES

('Epic Games', 'New York'),

('Speed Studios', 'San Francisco'),

('Mystic Games', 'London'),

('Zenith Publishers', 'Vancouver'),

('GameWorks', 'Tokyo');

7. Programmer Table

INSERT INTO Programmer (company_name, employee_id, name, specialization) VALUES

('Epic Games', 101, 'John Doe', 'Gameplay Developer'),

('Speed Studios', 102, 'Alice Smith', 'Backend Developer'),

('Mystic Games', 103, 'Bob Johnson', 'AI Specialist'),

('Zenith Publishers', 104, 'Sara Williams', 'Network Engineer'),

('GameWorks', 105, 'Kevin Brown', 'UI/UX Designer');

8. Artist Table

INSERT INTO Artist (company_name, employee_id, name, department) VALUES

('Epic Games', 201, 'Jane Miller', 'Concept Art'),

('Speed Studios', 202, 'Ethan Wilson', '3D Modeling'),

('Mystic Games', 203, 'Emily Davis', 'Animation'),

('Zenith Publishers', 204, 'Chris Martin', 'Visual Effects'),

('GameWorks', 205, 'Olivia Taylor', 'Character Design');

University of British Columbia, Vancouver
Department of Computer Science

9. Game Table

INSERT INTO Game (name, developing_company, release_year, genre, price, platform, publishing_company) VALUES

('Battle Royale', 'Epic Games', 2019, 'Shooter', 29.99, 'PC', 'Epic Games'),

('Fast Run', 'Speed Studios', 2021, 'Racing', 19.99, 'PC', 'Speed Studios'),

('Adventure Quest', 'Mystic Games', 2020, 'Adventure', 39.99, 'Xbox', 'Mystic Games'),

('War Zone', 'Epic Games', 2022, 'Shooter', 59.99, 'PlayStation', 'Epic Games'),

('Mystery Island', 'Mystic Games', 2023, 'Puzzle', 24.99, 'Switch', 'Mystic Games');