**CPSC 304 Project Cover Page**

Milestone #: \_\_\_2\_\_\_\_

Date: \_\_\_Jun 12, 2023\_\_\_

Group Number: \_\_\_\_\_9\_\_\_\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Student Number** | **CS Alias (Userid)** | **Preferred E-mail Address** |
| Abrielle Chan | 83047290 | r2o2n | abriellechan@gmail.com |
| Sahib Rao | 43038967 | m0z6r | sahibrao15@gmail.com |
| Kelcy Wang | 13162748 | d2l9o | kelcy.w.wang@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.

2. Summary

The domain of this application is Valorant players/users and their related information. This information includes skins that the user may have bought (including specific skins for specific weapons), what matches the users have played previously, what team the user plays for, and what agents the user played during those matches. The database models skins, weapons, matches, teams, game modes, maps, agents, and abilities. From this information, users can make conclusions like how many matches they have played, what agents they play the most, what game modes they play the most, what maps they tend to win on, what agents they tend to win with, etc.

3. ER Diagram

A picture containing diagram, sketch, plan, drawing

Description automatically generated

4. Schema

**Entities**

**Players**

(username: CHAR (20), rank: CHAR (20), acct level: INTEGER)

**Team**

(team\_name: CHAR (20), numOfMembers: INTEGER)

**Skins**

(skin\_name: CHAR (20), price: INTEGER)

**Guns**

(gun\_name: CHAR (20), c\_cost: INTEGER, type: CHAR (20))

**Match**

(ID: INTEGER, score: CHAR(5), duration: INTEGER, **gamemode\_name**: CHAR(20), **map\_name**: CHAR(20))

**GameMode**

(gamemode\_name: CHAR(20), numOfPlayers: INTEGER)

**Maps**

(map\_name: CHAR(20), numOfSites: INTEGER)

**Agents**

(agent\_name: CHAR (20), role: CHAR (20))  
  
**Abilities**

(ability\_name: CHAR (20), **agent\_name:** CHAR (20))  
  
**Passive**

(**ability\_name:** CHAR (20), **agent\_name:** CHAR (20))

**Active**

(**ability\_name:** CHAR (20), **agent\_name:** CHAR (20), c\_cost: INTEGER)

**Relationships**

**Owns**

(**username**: CHAR (20), **skin\_name**: CHAR (20))

**For**

(**skin\_name**: CHAR (20), **gun\_name**: CHAR (20))

**Plays**

(**username**: CHAR (20), **ID**: INTEGER, **agent\_name**: CHAR (20))

**Is On**

(**username:** CHAR(20) NOT NULL, **team\_name:** CHAR(20))

(not possible to convey total participation constraint with current knowledge – will use assertions for future milestones)

**Has** – see Ability table

**On, Is Type** – see Match table

5. Functional Dependencies

Username -> rank, acct\_level

Agent\_name -> role

Gun\_name -> c\_cost, gun\_type

Gamemode\_name -> NumPlayers

Match\_ID -> duration, score

Map\_name -> NumOfSites

Ability\_name -> c\_cost, Agent\_name

Skin\_name -> price

Team\_name -> NumMembers

score -> duration

A: Username

B: rank

C: acct\_level

D: Agent\_name

E: role

F: Gun\_name

G: c\_cost

H: gun\_type

I: Gamemode\_name

J: duration

K: NumPlayers

L: Match\_ID

M: score

N: Map\_name

O: NumOfSites

P: Ability\_name

Q: c\_cost

R: Skin\_name

S: price

T: Team\_name

U: NumMembers

A -> B, C

D -> E

F -> G, H

I -> K

L -> J, M

N -> O

P -> Q, D

R -> S

T -> U

M -> J

Minimal Cover:

A -> B

A -> C

D -> E

F -> G

F -> H

I -> K

L -> M

N -> O

P -> Q

P -> D

R -> S

T -> U

M -> J

(Minimal Keys): AFILNPRT

V(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U)

6. Decomposition

**Players**

(username: CHAR (20), rank: CHAR (20), acct level: INTEGER)

Players(A, B, C)

Relevant FDs:

A->B

A->C

CKs: A

Both FDs are in BCNF (and 3NF) since A is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Team**

(team\_name: CHAR (20), numOfMembers: INTEGER)

Team(T, U)

Relevant FDs:

T->U

CKs: T

T->U is in BCNF (and 3NF) since T is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Skins**

(skin\_name: CHAR (20), price: INTEGER)

Skins(R, S)

Relevant FDs:

R->S

CKs: R

R->S is in BCNF (and 3NF) since R is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Guns**

(gun\_name: CHAR (20), c\_cost: INTEGER, type: CHAR (20))

Guns(F, G, H)

Relevant FDs:

F->G

F->H

CKs: F

Both FDs are in BCNF (and 3NF) since F is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Match**

(ID: INTEGER, score: CHAR(5), duration: INTEGER, **gamemode\_name**: CHAR(20), **map\_name**: CHAR(20))

Match(L, M, J, I, N)

Relevant FDs:

L -> M

M -> J

(already in minimal cover from #5)

Find key:

Left: I, L, N

Middle: M

Right: J

CKs: ILN

Check FDs:

L->M

L is not a superkey, but M is not part of the CK -> fails 3NF

M->J

M is not a superkey, and J is not part of the CK -> fails 3NF

Decompose using M->J

M1(MJ), M2(LINM)

Check M1:

Relevant FDs:

M->J

M1 has two attributes, which puts it in BCNF (and 3NF as well).

Check M2:

Relevant FDs:

L->M

Find keys:

Left: L, N, I

Middle:

Right: M

CKs: LNI

Check FDs:

L->M

L is not a superkey, M is not a part of the CK

Decompose using L->M

M3(LM), M4(LNI)

Check M3:

Relevant FDs:

L->M

M3 has two attributes, which puts it in BCNF (and 3NF as well)

Check M4:

Relevant FDs:

None

LNI is the CK, so M4 is in BCNF

Final decomposition: M1(MJ), M3(LM), M4(LNI)

M1(score, duration)

M3(ID, score)

M4(ID, Map\_name, Gamemode\_name)

**GameMode**

(gamemode\_name: CHAR(20), numOfPlayers: INTEGER)

GameMode(I, K)

Relevant FDs:

I->K

CKs: I

I->K is in BCNF (and 3NF) since I is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Maps**

(map\_name: CHAR(20), numOfSites: INTEGER)

Maps(N, O)

Relevant FDs:

N->O

CKs: N

N->O is in BCNF (and 3NF) since N is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Agents**

(agent\_name: CHAR (20), role: CHAR (20))

Agents(D, E)

Relevant FDs:

D->E

CKs: D

D->E is in BCNF (and 3NF) since D is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Abilities**

(ability\_name: CHAR (20), **agent\_name:** CHAR (20))

Abilities(P, D)

Relevant FDs:

P->D

CKs: P

P->D is in BCNF (and 3NF) since P is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.   
  
**Passive**

(**ability\_name:** CHAR (20), **agent\_name:** CHAR (20))

Passive(P, D)

Relevant FDs:

P->D

CKs: P

P->D is in BCNF (and 3NF) since P is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Active**

(**ability\_name:** CHAR (20), **agent\_name:** CHAR (20), c\_cost: INTEGER)

Active(P, D, Q)

Relevant FDs:

P->D

P->Q

CKs: P

Both FDs are in BCNF (and 3NF) since P is a candidate key (which makes it a superkey as well.)

This relationship does not need to be decomposed.

**Owns**

(**username**: CHAR (20), **skin\_name**: CHAR (20))

Owns(A, R)

Relevant FDs:

None

CKs: AR

There are no non-trivial FDs to be violated – this relationship is in BCNF (and 3NF.)

This relationship does not need to be decomposed.

**For**

(**skin\_name**: CHAR (20), **gun\_name**: CHAR (20))

For(R, F)

Relevant FDs:

None

CKs: RF

There are no non-trivial FDs to be violated – this relationship is in BCNF (and 3NF.)

This relationship does not need to be decomposed.

**Plays**

(**username**: CHAR (20), **ID**: INTEGER, **agent\_name**: CHAR (20))

Plays(A, L, D)

Relevant FDs:

None

CKs: ALD

There are no non-trivial FDs to be violated – this relationship is in BCNF (and 3NF.)

This relationship does not need to be decomposed.

**Is On**

(**username:** CHAR(20) NOT NULL, **team\_name:** CHAR(20))

IsOn(A, T)

Relevant FDs:

None

CKs: AT

There are no non-trivial FDs to be violated – this relationship is in BCNF (and 3NF.)

This relationship does not need to be decomposed.

**Has** – see Ability table

**On, Is Type** – see Match table

7. CREATE TABLE Statements

CREATE TABLE players(

username CHAR (20),

rank CHAR (20),

acct\_level INTEGER,

PRIMARY KEY (username)

);

CREATE TABLE team(

team\_name CHAR (20),

num\_of\_members CHAR (20),

PRIMARY KEY (team\_name)

);

CREATE TABLE skin(

skin\_name CHAR (20),

price INTEGER,

PRIMARY KEY (skin\_name)

);

CREATE TABLE gun(

gun\_name CHAR (20),

c\_cost INTEGER,

type CHAR(20)

PRIMARY KEY (gun\_name)

);

CREATE TABLE match(

match\_ID INTEGER,

gamemode\_name CHAR (20) NOT NULL,

Map\_name CHAR (20) NOT NULL,

PRIMARY KEY (match\_ID),

FOREIGN KEY (gamemode\_name) REFERENCES gamemode,

ON DELETE CASCADE,

ON UPDATE CASCADE

FOREIGN KEY (map\_name) REFERENCES map);

ON DELETE CASCADE,

ON UPDATE CASCADE

);

CREATE TABLE gamemode(

gamemode\_name CHAR (20),

num\_of\_players INTEGER,

PRIMARY KEY (gamemode\_name)

);

CREATE TABLE map(

map\_name CHAR (20),

numOfSites INTEGER,

PRIMARY KEY (map\_name)

);

CREATE TABLE agent(

agent\_name CHAR (20),

role CHAR(20),

PRIMARY KEY (agent\_name)

);

CREATE TABLE ability(

ability\_name CHAR (20),

agent\_name CHAR(20) NOT NULL,

PRIMARY KEY (ability\_name, agent\_name),

FOREIGN KEY (agent\_name) REFERENCES agent(agent\_name),

ON DELETE CASCADE,

ON UPDATE CASCADE

);

CREATE TABLE passive(

ability\_name CHAR (20),

agent\_name CHAR(20),

PRIMARY KEY (ability\_name, agent\_name),

FOREIGN KEY (agent\_name) REFERENCES agent(agent\_name),

ON DELETE CASCADE,

ON UPDATE CASCADE,

FOREIGN KEY (ability\_name) REFERENCES ability(ability\_name),

ON DELETE CASCADE,

ON UPDATE CASCADE

);

CREATE TABLE active(

ability\_name CHAR (20),

agent\_name CHAR(20),

c\_cost INTEGER,

PRIMARY KEY (ability\_name, agent\_name),

FOREIGN KEY (agent\_name) REFERENCES agent(agent\_name),

ON DELETE CASCADE,

ON UPDATE CASCADE,

FOREIGN KEY (ability\_name) REFERENCES ability(ability\_name),

ON DELETE CASCADE,

ON UPDATE CASCADE

);

CREATE TABLE owns (

username CHAR(20),

gun\_name, CHAR(20),

PRIMARY KEY (username , gun\_name),

FOREIGN KEY (username ) REFERENCES player,

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIGN KEY (gun\_name) REFERENCES gun,

ON DELETE CASCADE

ON UPDATE CASCADE

);

CREATE TABLE for (

skin\_name CHAR(20),

gun\_name, CHAR(20),

PRIMARY KEY (skin\_name , gun\_name),

FOREIGN KEY (skin\_name) REFERENCES skin,

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIGN KEY (gun\_name) REFERENCES gun,

ON DELETE CASCADE

ON UPDATE CASCADE

);

CREATE TABLE plays (

username CHAR(20),

match\_ID, INTEGER,

agent name, CHAR(20),

PRIMARY KEY (username, match\_ID, agent\_name),

FOREIGN KEY (username) REFERENCES player,

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIGN KEY (match\_ID) REFERENCES match,

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIGN KEY (agent\_name) REFERENCES agent,

ON DELETE CASCADE

ON UPDATE CASCADE

);

CREATE TABLE isOn (

username CHAR(20),

team\_name CHAR(20)

);

8. INSERT Statements

**Players**

INSERT INTO Players (username, rank, acct level, team\_name)

VALUES (‘namoraeh#rae’, ‘Silver\_II’, 253);

INSERT INTO Players (username, rank, acct level, team\_name)

VALUES (‘mango#tango’, ‘Platinum\_I’, 356);

INSERT INTO Players (username, rank, acct level, team\_name)

VALUES (‘blueberry#xiao’, ‘Silver\_II’, 280);

INSERT INTO Players (username, rank, acct level, team\_name)

VALUES (‘strawberry#3864’, ‘Gold\_III’, 86’);

INSERT INTO Players (username, rank, acct level, team\_name)

VALUES (‘spaghetti#6969’, ‘Iron\_III’, 45);

**Team**

INSERT INTO Team (team\_name, numOfMembers)

VALUES (‘Fruits’, 5);

INSERT INTO Team (team\_name, numOfMembers)

VALUES (‘Fruits 2.0’, 6);

INSERT INTO Team (team\_name, numOfMembers)

VALUES (‘Road to Iron’, 5);

INSERT INTO Team (team\_name, numOfMembers)

VALUES (‘Dream Team’, 7);

INSERT INTO Team (team\_name, numOfMembers)

VALUES (‘Team Name Pending’, 5);

**Skins**

INSERT INTO Skins (skin\_name, price)

VALUES (‘Reaver’, 1775);

INSERT INTO Skins (skin\_name, price)

VALUES (‘Spectrum’, 2675);

INSERT INTO Skins (skin\_name, price)

VALUES (‘Silvanus’, 1275);

INSERT INTO Skins (skin\_name, price)

VALUES (‘Prime’, 1775);

INSERT INTO Skins (skin\_name, price)

VALUES (‘Smite’, 875);

**Guns**

INSERT INTO Guns (gun\_name, c\_cost, type)

VALUES (‘Vandal’, 2900, ‘Rifle’);

INSERT INTO Guns (gun\_name, c\_cost, type)

VALUES (‘Spectre’, 1600, ‘SMG’);

INSERT INTO Guns (gun\_name, c\_cost, type)

VALUES (‘Frenzy’, 450, ‘Pistol’);

INSERT INTO Guns (gun\_name, c\_cost, type)

VALUES (‘Judge’, 1850, ‘Shotgun’);

INSERT INTO Guns (gun\_name, c\_cost, type)

VALUES (‘Operator’, 4700, ‘Sniper’);

**Match**

INSERT INTO Match (ID, score, duration, gamemode\_name, map\_name)

VALUES (12973, ‘13-6’, 45, ‘Unrated’, ‘Lotus’);

INSERT INTO Match (ID, score, duration, gamemode\_name, map\_name)

VALUES (12974, ‘10-13’, 60, ‘Competitive’, ‘Bind’);

INSERT INTO Match (ID, score, duration, gamemode\_name, map\_name)

VALUES (12975, ‘4-1’, 10, ‘Spike Rush’, ‘Icebox’);

INSERT INTO Match (ID, score, duration, gamemode\_name, map\_name)

VALUES (12976, ‘22-40’, 10, ‘Deathmatch’, ‘Breeze’);

INSERT INTO Match (ID, score, duration, gamemode\_name, map\_name)

VALUES (12977, ‘5-4’, 15, ‘Swiftplay’, ‘Ascent’);

**GameMode**

INSERT INTO GameMode (gamemode\_name, duration, numOfPlayers)

VALUES (‘Unrated’, 10);

INSERT INTO GameMode (gamemode\_name, duration, numOfPlayers)

VALUES (‘Competitive’, 10);

INSERT INTO GameMode (gamemode\_name, duration, numOfPlayers)

VALUES (‘Swiftplay’, 10);

INSERT INTO GameMode (gamemode\_name, duration, numOfPlayers)

VALUES (‘Spike Rush’, 10);

INSERT INTO GameMode (gamemode\_name, duration, numOfPlayers)

VALUES (‘Deathmatch’, 14);

**Maps**

INSERT INTO Maps (map\_name, numOfSites)

VALUES (‘Lotus’, 3);

INSERT INTO Maps (map\_name, numOfSites)

VALUES (‘Bind’, 2);

INSERT INTO Maps (map\_name, numOfSites)

VALUES (‘Haven’, 3);

INSERT INTO Maps (map\_name, numOfSites)

VALUES (‘Ascent’, 2);

INSERT INTO Maps (map\_name, numOfSites)

VALUES (‘Icebox’, 2);

**Agents**

INSERT INTO Agents (agent\_name, role)

VALUES (‘Breach’, ‘Initiator’);

INSERT INTO Agents (agent\_name, role)

VALUES (‘Sage’, ‘Sentinel’);

INSERT INTO Agents (agent\_name, role)

VALUES (‘Viper’, ‘Controller’);

INSERT INTO Agents (agent\_name, role)

VALUES (‘Phoenix’, ‘Duelist’);

INSERT INTO Agents (agent\_name, role)

VALUES (‘Reyna’, ‘Duelist’);

**Abilities - Passive**

INSERT INTO Passive (ability\_name, agent\_name)

VALUES (‘Drift’, Jett);

INSERT INTO Passive (ability\_name, agent\_name)

VALUES (‘Heal’, Phoenix);

INSERT INTO Passive (ability\_name, agent\_name)

VALUES (‘Soul Orb’, Reyna);

INSERT INTO Passive (ability\_name, agent\_name)

VALUES (‘Buddy Orb’, Gekko);

INSERT INTO Passive (ability\_name, agent\_name)

VALUES (‘Terror Trail’, Fade);

**Abilities - Active**

INSERT INTO Active (ability\_name, agent\_name, c\_cost)

VALUES (‘Flashpoint’, ‘Breach’, 250);

INSERT INTO Active (ability\_name, agent\_name, c\_cost)

VALUES (‘Hot Hands’, ‘Phoenix’, 0);

INSERT INTO Active (ability\_name, agent\_name, c\_cost)

VALUES (‘Snakebite’, ‘Viper’, 200);

INSERT INTO Active (ability\_name, agent\_name, c\_cost)

VALUES (‘Barrier Orb’, ‘Sage’, 400);

INSERT INTO Active (ability\_name, agent\_name, c\_cost)

VALUES (‘Leer’, ‘Reyna’, 250);

**Owns**

INSERT INTO Owns (username, skin\_name)

VALUES (‘namoraeh#rae’, ‘Reaver\_Vandal’);

INSERT INTO Owns (username, skin\_name)

VALUES (‘mango#tango, ‘Spectrum\_Phantom’);

INSERT INTO Owns (username, skin\_name)

VALUES (‘mango#tango, ‘Prime\_Frenzy’);

INSERT INTO Owns (username, skin\_name)

VALUES (‘strawberry#3864’, ‘Prime\_Frenzy’);

INSERT INTO Owns (username, skin\_name)

VALUES (‘strawberry#3864’, ‘Silvanus\_Stinger’);

**For**

INSERT INTO For (skin\_name, gun\_name)

VALUES (‘Reaverl’, ‘Vandal’);

INSERT INTO For (skin\_name, gun\_name)

VALUES (‘Spectrum’, ‘Phantom’);

INSERT INTO For (skin\_name, gun\_name)

VALUES (‘Prime’, ‘Frenzy’);

INSERT INTO For (skin\_name, gun\_name)

VALUES (‘Silvanus’, ‘Stinger’);

INSERT INTO For (skin\_name, gun\_name)

VALUES (‘Smite’, ‘Classic’);

**Plays**

INSERT INTO Plays (username, ID, agent\_name)

VALUES (‘namoraeh#rae’, 12973, ‘Breach’);

INSERT INTO Plays (username, ID, agent\_name)

VALUES (‘mango#tango’, 12974, ‘Reyna’);

INSERT INTO Plays (username, ID, agent\_name)

VALUES (‘mango#tango’, 12975, ‘Reyna’);

INSERT INTO Plays (username, ID, agent\_name)

VALUES (‘strawberry#3864’, 12976, ‘Sage’);

INSERT INTO Plays (username, ID, agent\_name)

VALUES (‘spaghetti#6969’, 12976, ‘Gekko’);

**IsOn**

INSERT INTO Is\_On (username, team\_name)

VALUES (‘mango#tango’, ‘Fruits’);

INSERT INTO Is\_On (username, team\_name)

VALUES (‘pine#apple’, ‘Fruits’);

INSERT INTO Is\_On (username, team\_name)

VALUES (‘strawberry#3864’, ‘Fruits 2.0’);

INSERT INTO Is\_On (username, team\_name)

VALUES (‘spaghetti#6969’, ‘Road to Iron’);

INSERT INTO Is\_On (username, team\_name)

VALUES (‘isuckatthisgame#help’, ‘Road to Iron’);