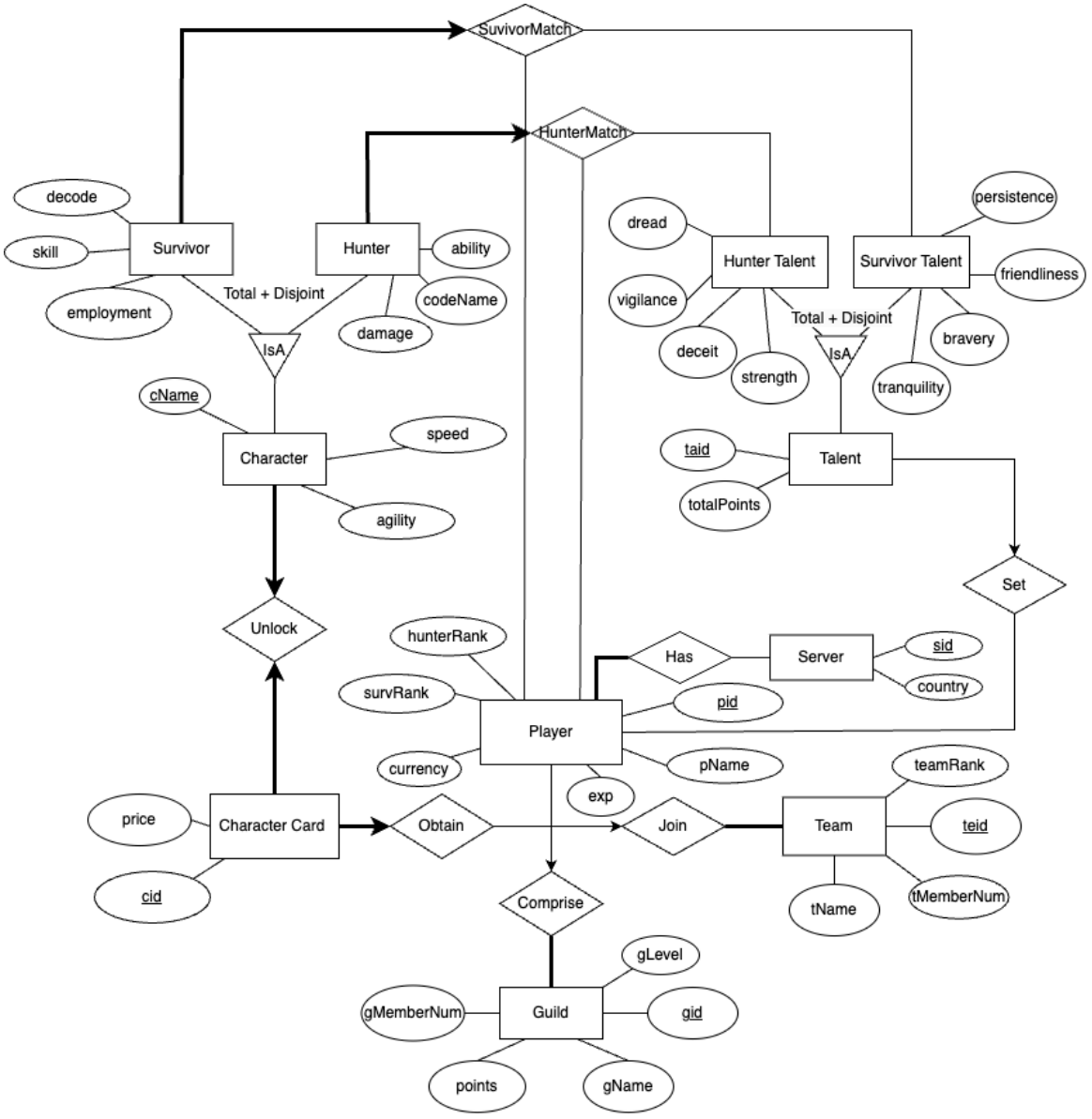
2. Brief summary

This project develops a player profile and game character database for Identity V, an asymmetrical survival horror game. It tracks **players**, their **server**, and **character cards**, which unlock **characters** (survivors and hunters). The database also includes **guilds**, **teams**, and **talents** that enhance character abilities.

3. Changes made

a. Added overlap/covering constraints for ISA
b. changed Player's attribute Name to pName to align with other entityName
changed CharacterCard's attribute ccName into cid
c. changed player set talent from many to many to one to many relationship;
d. changed relationships “Give” and “Attach” to “SuvivorMatch” and “HunterMatch”; each added player to form a ternary relationship; added constraint that each hunter or survivor must match to exact 1 HunterTalent or SuvivorTalent
e. Deleted relation “Pick” between Character and Player



4. Schema (Blue-marked could be null, the others are not null)

Character (cName: varchar[20], speed: integer, agility: integer)
Survivor (**cName**: varchar[20], skill: varchar[20], decode: integer, employment: varchar[20])
Hunter (**cName**: varchar[20], ability: varchar[20], damage: integer, codeName: varchar[20])
Guild (gid: integer, gName (unique, CK): varchar[20], gLevel: integer, gMemberNum: integer, points: integer)
Team (teid: integer, tName (unique, CK): varchar[20], teamRank: integer, tMemberNum: integer)
Server (sid: integer, country: varchar[20])
Player (pid: integer, pName (unique, CK): varchar[20], exp: integer, survRank: integer, hunterRank: integer, currency: integer, **sid**: integer, **gid**: integer, **teid**: integer)
CharacterCard (cid: integer, price: integer, **cName**: varchar[20], **pid**: integer)
Talent (taid: integer, totalPoints: integer, **pid**: integer)
SurvivorTalent (**taid**: integer, friendliness: integer, bravery: integer, tranquility: integer, persistence: integer)
HunterTalent (**taid**: integer, dread: integer, vigilance: integer, deceit: integer, strength: integer)
SuvivorMatch (**pid**: integer, **cName**: varchar[20], **taid**: integer)
HunterMatch (**pid**: integer, **cName**: varchar[20], **taid**: integer)

5. FDs

Character (cName -> speed agility)
Survivor (cName -> employment, employment -> decode skill)
Hunter (cName -> codeName, codeName -> damage ability)
Guild (gid -> gName gLevel gMemberNum Points)
Team (teid -> tName teamRank tMemberNum)
Server (sid -> country)
Player (pid -> pName exp survRank hunterRank currency sid gid teid)
CharacterCard (cid -> cName pid, cName -> price)
Talent (taid -> totalPoints pid)
HunterTalent (taid -> dread vigilance deceit strength)
SuvivorTalent (taid -> friendliness bravery tranquility persistence)
SurvivorMatch (cName pid -> taid)
HunterMatch (cName pid -> taid)

6. Normalization: We want to normalize into 3NF

Character: Already in 3NF, because both speed and agility depend only on primary key cName.

Character (cName, speed, agility)

Survivor: There is a transitive dependency: employment → decode skill. So we need to split the relation into two, with employment being the primary key for the second relation

SurvivorStats (employment, decode, skill)
Survivor (**cName**, **employment**)

Hunter: There is a transitive dependency: codeName → damage ability. We need to split the relation into two, with codeName being the primary key for the second relation

HunterStats (codeName, damage, ability)
Hunter (**cName**, **codeName**)

Guild: Already in 3NF, because all attributes (gName, gLevel, gMemberNum, points) depend on primary key gid.

Guild (gid, gName, gLevel, gMemberNum, points)

Team: Already in 3NF, because all attributes (tName, teamRank, tMemberNum) depend on primary key teid.

Team (teid, tName, teamRank, tMemberNum)

Server: Already in 3NF, as country depends on primary key sid.

Server (sid, country)

Player: Already in 3NF, because all attributes (pName, exp, survRank, hunterRank, currency, sid, gid, teid) depend on primary key pid.

Player (pid, pName, exp, survRank, hunterRank, currency, sid, gid, teid)

CharacterCard: Partial dependency cName → price. We need to split the relation into two, with cName being the primary key for the second relation

CardPrice (cName, price)
CardOwnership (cid, **cName**, pid)

Talent: Already in 3NF, because (totalPoints, pid) depend on primary key taid.

Talent (taid, totalPoints, **pid**)

HunterTalent: Already in 3NF, as all attributes (dread, vigilance, deceit, strength) depend on primary key taid.

HunterTalent (**taid**, dread, vigilance, deceit, strength)

SurvivorTalent: Already in 3NF, as all attributes (friendliness, bravery, tranquility, persistence) depend on primary key taid.

SurvivorTalent (**taid**, friendliness, bravery, tranquility, persistence)

SurvivorMatch Relation: Already in 3NF, as the composite primary key (cName, pid) determines taid.

SurvivorMatch (**cName**, **pid**, **taid**)

HunterMatch Relation: Already in 3NF, as the composite primary key (cName, pid) determines taid.

HunterMatch (**cName**, **pid**, **taid**)

Final Relations in 3NF:

Character (cName, speed, agility)
SurvivorStats (employment, decode, skill)
Survivor (**cName**, **employment**)
HunterStats (codeName, damage, ability)
Hunter (**cName**, **codeName**)
Guild (gid, gName, gLevel, gMemberNum, points)
Team (teid, tName, teamRank, tMemberNum)
Server (sid, country)
Player (pid, pName, exp, survRank, hunterRank, currency, **sid, gid, teid**)
CardPrice (cName, price)
CardOwnership (cid, **cName**, pid)
Talent (taid, totalPoints, **pid**)
HunterTalent (**taid**, dread, vigilance, deceit, strength)
SurvivorTalent (**taid**, friendliness, bravery, tranquility, persistence)
SurvivorMatch (**cName**, **pid**, **taid**)
HunterMatch (**cName**, **pid**, **taid**)

*Note all CKs remain the same from the previous step

7. SQL DDL

```
CREATE TABLE Character (
    cName VARCHAR(20) PRIMARY KEY,
    speed INT NOT NULL,
    agility INT NOT NULL
);

CREATE TABLE SurvivorStats (
    employment VARCHAR(20) PRIMARY KEY,
    decode INT NOT NULL,
    skill VARCHAR(20) NOT NULL
);

CREATE TABLE Survivor (
    cName VARCHAR(20) PRIMARY KEY,
    employment VARCHAR(20) NOT NULL,
    FOREIGN KEY (cName) REFERENCES Character(cName) ON DELETE CASCADE,
    FOREIGN KEY (employment) REFERENCES SurvivorStats(employment) ON DELETE
CASCADE
);

CREATE TABLE HunterStats (
    codeName VARCHAR(20) PRIMARY KEY,
    damage INT NOT NULL,
    ability VARCHAR(20) NOT NULL
);

CREATE TABLE Hunter (
    cName VARCHAR(20) PRIMARY KEY,
    codeName VARCHAR(20) NOT NULL,
    FOREIGN KEY (cName) REFERENCES Character (cName) ON DELETE CASCADE,
    FOREIGN KEY (codeName) REFERENCES HunterStats (codeName) ON DELETE
CASCADE
    );

CREATE TABLE CardPrice (
    cName VARCHAR(20) PRIMARY KEY,
    price INT NOT NULL,
    FOREIGN KEY (cName) REFERENCES Character (cName) ON DELETE CASCADE
);

CREATE TABLE CardOwnership (
    cid INT PRIMARY KEY,
    cName VARCHAR(20) NOT NULL,
    pid INT NOT NULL,
    FOREIGN KEY (cName) REFERENCES CardPrice (cName) ON DELETE CASCADE,
    FOREIGN KEY (pid) REFERENCES Player (pid) ON DELETE CASCADE
```

```
);

CREATE TABLE Guild (
    gid INT PRIMARY KEY,
    gName VARCHAR(20) NOT NULL,
    gLevel INT NOT NULL,
    gMemberNum INT NOT NULL,
    points INT NOT NULL
);

CREATE TABLE Team (
    teid INT PRIMARY KEY,
    tName VARCHAR(20) NOT NULL,
    teamRank INT NOT NULL,
    tMemberNum INT NOT NULL
);

CREATE TABLE Server (
    sid INT PRIMARY KEY,
    country VARCHAR(20) NOT NULL
);

CREATE TABLE Player (
    pid INT PRIMARY KEY,
    pName VARCHAR(20) NOT NULL,
    exp INT NOT NULL,
    survRank INT NOT NULL,
    hunterRank INT NOT NULL,
    currency INT NOT NULL,
    sid INT NOT NULL,
    gid INT,
    teid INT,
    FOREIGN KEY (sid) REFERENCES Server(sid) ON DELETE CASCADE,
    FOREIGN KEY (gid) REFERENCES Guild(gid) ON DELETE CASCADE,
    FOREIGN KEY (teid) REFERENCES Team(teid) ON DELETE CASCADE
);

# totalPoints <= 120
CREATE TABLE Talent (
    taid INT PRIMARY KEY,
    totalPoints INT NOT NULL,
    pid INT NOT NULL,
    FOREIGN KEY (pid) REFERENCES Player(pid) ON DELETE CASCADE
);

# dread, vigilance, deceit, strength each has max of 40
CREATE TABLE HunterTalent (
    taid INT PRIMARY KEY,
```

```
    dread INT DEFAULT 0,
    vigilance INT DEFAULT 0,
    deceit INT DEFAULT 0,
    strength INT DEFAULT 0,
    FOREIGN KEY (taid) REFERENCES Talent(taid) ON DELETE CASCADE
);

# friendliness, bravery, tranquility, persistence each has max of 40
CREATE TABLE SurvivorTalent (
    taid INT PRIMARY KEY,
    friendliness INT DEFAULT 0,
    bravery INT DEFAULT 0,
    tranquility INT DEFAULT 0,
    persistence INT DEFAULT 0,
    FOREIGN KEY (taid) REFERENCES Talent(taid) ON DELETE CASCADE
);

CREATE TABLE SurvivorMatch (
    cName VARCHAR(20),
    pid INT NOT NULL,
    taid INT NOT NULL,
    PRIMARY KEY (cName, pid),
    FOREIGN KEY (cName) REFERENCES Character(cName) ON DELETE CASCADE,
    FOREIGN KEY (pid) REFERENCES Player(pid) ON DELETE CASCADE,
    FOREIGN KEY (taid) REFERENCES Talent(taid) ON DELETE CASCADE
);

CREATE TABLE HunterMatch (
    cName VARCHAR(20) NOT NULL,
    pid INT NOT NULL,
    taid INT NOT NULL,
    PRIMARY KEY (cName, pid),
    FOREIGN KEY (cName) REFERENCES Character(cName) ON DELETE CASCADE,
    FOREIGN KEY (pid) REFERENCES Player(pid) ON DELETE CASCADE,
    FOREIGN KEY (taid) REFERENCES Talent(taid) ON DELETE CASCADE
);
```

8. INSERT statements

```
INSERT INTO Character(cName, speed, agility)
VALUES
('Freddy Riley', 263, 73),
('Emily Dyer', 263, 73),
('Kreacher Pierson', 263, 73),
('Emma Woods', 263, 73),
('Tracy Reznik', 263, 86),
('Jeffrey Bonavita', 212, 38),
('Leo', 216, 35),
('Joker', 219, 25),
('Grace', 216, 27),
('Ivy', 216, 35);

INSERT INTO SurvivorStats(employment, decode, skill)
VALUES
('Lawyer', 70, 'Foresight'),
('Doctor', 81, 'Med Master'),
('Thief', 81, 'Cunning'),
('Gardener', 81, 'Ingenuity'),
('Mechanic', 74, 'Operator');

INSERT INTO Survivor(cName, employment)
VALUES
('Freddy Riley', 'Lawyer'),
('Emily Dyer', 'Doctor'),
('Kreacher Pierson', 'Thief'),
('Emma Woods', 'Gardener'),
('Tracy Reznik', 'Mechanic');

INSERT INTO HunterStats(codeName, damage, ability)
VALUES
('Goatman', 25, 'Shackled'),
('Hell Ember', 50, 'Infernal Soul'),
('Smiley Face', 50, 'Rocket Modification'),
('Naiad', 50, 'Darkest Depths'),
('The Shadow', 50, 'Phantasm');

INSERT INTO Hunter(cName, codeName)
VALUES
('Jeffrey Bonavita', 'Goatman'),
('Leo', 'Hell Ember'),
('Joker', 'Smiley Face'),
('Grace', 'Naiad'),
('Ivy', 'The Shadow')

INSERT INTO Guild (gid, gName, gLevel, gMemberNum, points)
```

```
VALUES
(1, 'GuildOne', 5, 1, 500),
(2, 'GuildTwo', 4, 1, 400),
(3, 'GuildThree', 6, 2, 600),
(4, 'GuildFour', 7, 1, 700),
(5, 'GuildFive', 3, 1, 300);

INSERT INTO Team (teid, tName, teamRank, tMemberNum)
VALUES
(1, 'TeamOne', 1, 1),
(2, 'TeamTwo', 2, 1),
(3, 'TeamThree', 3, 1),
(4, 'TeamFour', 4, 1),
(5, 'TeamFive', 5, 2);

INSERT INTO Server (sid, country)
VALUES
(1, 'USA'),
(2, 'Canada'),
(3, 'Japan'),
(4, 'Korea'),
(5, 'Germany');

INSERT INTO Player (pid, pName, exp, survRank, hunterRank, currency, sid, gid, teid)
VALUES
(1001, 'Lucia', 10, 7, 7, 1000, 1, 1, 1),
(1002, 'Simon', 98, 1, 1, 1500, 2, 2, 2),
(1003, 'Akira', 50, 1, 1, 800, 3, 3, 3),
(1004, 'Player4', 130, 5, 6, 2000, 4, 4, 4),
(1005, 'Player5', 140, 4, 3, 2500, 5, 5, 5),
(1006, 'Player6', 200, 6, 7, 3000, 1, 3, 5);

INSERT INTO CardPrice (cName, price)
VALUES
('Emily Dyer', 488),
('Freddy Riley', 488),
('Jeffrey Bonavita', 688),
('Grace', 688),
('Tracy Reznik', 488);

INSERT INTO CardOwnership (cid, cName, pid)
VALUES
(1, 'Emily Dyer', 1001),
(2, 'Freddy Riley', 1001),
(3, 'Jeffrey Bonavita', 1001),
(4, 'Grace', 1002),
(5, 'Tracy Reznik', 1005);
```

```
INSERT INTO Talent (taid, totalPoints, pid)
VALUES
(1, 90, 1001),
(2, 120, 1002),
(3, 120, 1003),
(4, 120, 1004),
(5, 120, 1005),
(6, 90, 1001),
(7, 120, 1002),
(8, 120, 1003),
(9, 120, 1004),
(10, 120, 1005);

INSERT INTO HunterTalent (taid, dread, vigilance, deceit, strength)
VALUES
(1, 30, 30, 0, 30),
(2, 40, 40, 0, 40),
(3, 10, 30, 40, 40),
(4, 30, 30, 30, 30),
(5, 15, 25, 40, 40);

INSERT INTO SurvivorTalent (taid, friendliness, bravery, tranquility, persistence)
VALUES
(6, 30, 10, 20, 30),
(7, 20, 20, 40, 40),
(8, 40, 40, 40, 0),
(9, 25, 25, 35, 35),
(10, 10, 30, 40, 40);

INSERT INTO SurvivorMatch (cName, pid, taid)
VALUES
('Freddy Riley', 1001, 1),
('Emily Dyer', 1001, 1),
('Emily Dyer', 1002, 2),
('Kreacher Pierson', 1003, 3),
('Emma Woods', 1004, 4),
('Tracy Reznik', 1005, 5);

INSERT INTO HunterMatch (cName, pid, taid)
VALUES
('Jeffrey Bonavita', 1001, 6),
('Leo', 1002, 7),
('Joker', 1003, 8),
('Grace', 1004, 9),
('Ivy', 1005, 10);
```