University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

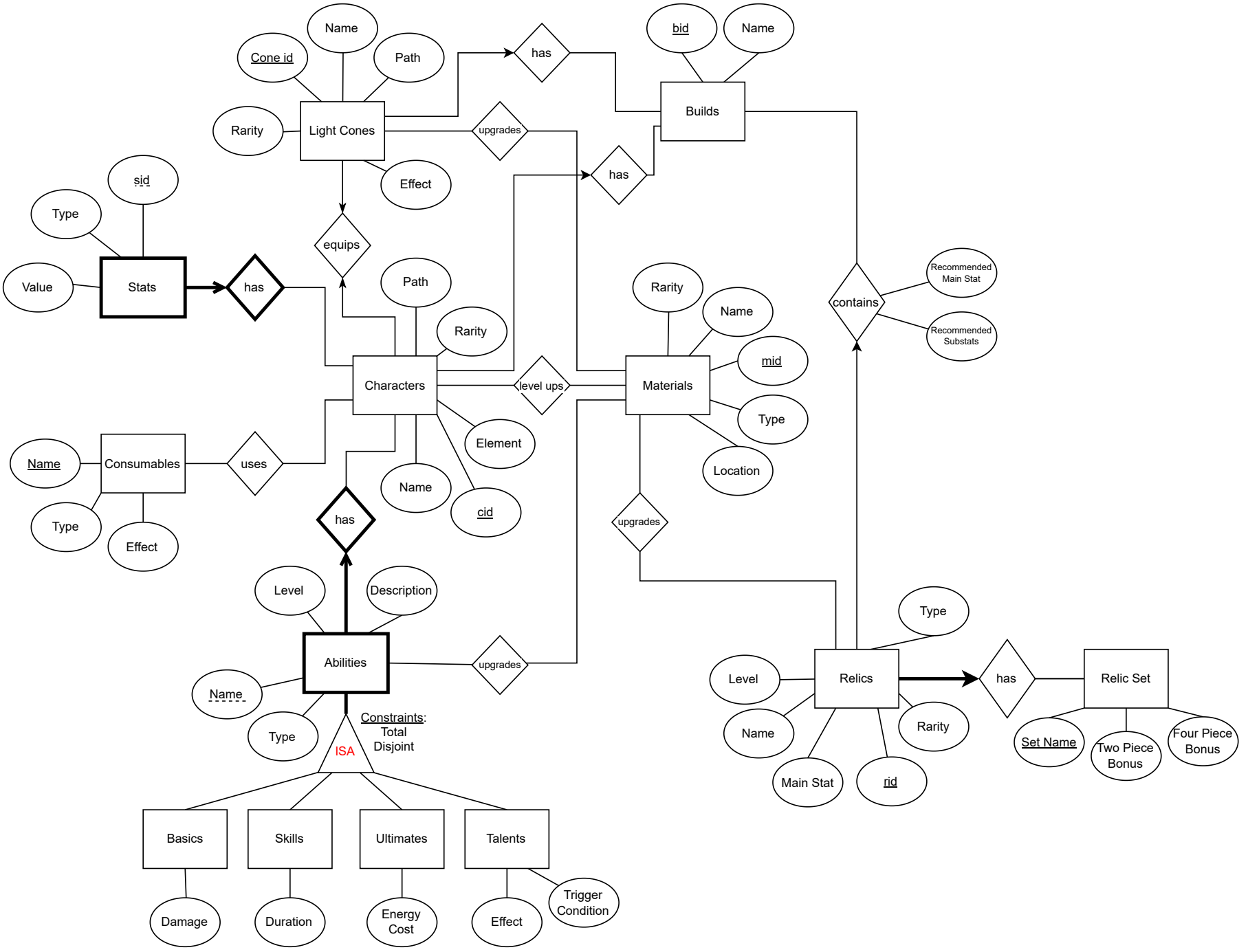
Date: Mar 3, 2025

Group Number: 42

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Angela Li	51659845	x4o7u	ali151@student.ubc.ca
Stephanie Leung	54288568	w0t6f	sleung32@student.ubc.ca
Tyler Yih	71046452	f7d3s	tyih985@student.ubc.ca

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


Name
Cone id
Path
has
bid
Name
Builds
Rarity
Light Cones
upgrades
Effect
has
sid
Type
equips
Value
Stats
has
Path
Rarity
Rarity
Name
Recommended Main Stat
contains
Recommended Substats
Characters
level ups
Materials
mid
Type
Element
Location
Name
Consumables
uses
Name
cid
Type
Effect
has
upgrades
Level
Description
Type
Relics
has
Relic Set
Abilities
upgrades
Level
Name
Rarity
Name
Set Name
Two Piece Bonus
Four Piece Bonus
Type
Constraints: Total Disjoint
ISA
Main Stat
rid
Basics
Skills
Ultimates
Talents
Damage
Duration
Energy Cost
Effect
Trigger Condition

Milestone 2

Summary

Our project is focused on the game, Honkai: Star Rail. More specifically, it aims to help users with character builds, storing information about what a character can equip like light cones and relics, along with the associated materials needed. Additionally, it also stores character stats and abilities.

ER Diagram Changes

In the ER diagram, we fixed the dashed underlining required for weak entity's partial keys. Additionally, we changed the relationship between light cone and builds to be a one to many relationship instead of a one-to-one to facilitate the same recommended light cone for multiple characters if user has multiple copies of the same light cone. The `character id` attribute from Abilities entity has also been removed to be used as a foreign key from relation instead. In the context of our project, we also changed the enforcement of total participation between relics and relic sets such that every relic must have a set rather than the other way around, so that when users delete relics, information about the set isn't also deleted.

Schema

Characters(cid: INTEGER, name: VARCHAR, element: VARCHAR, rarity: INTEGER, path: VARCHAR, cone_id: INTEGER, bid: INTEGER)

- cid is the PK
- cone_id is a FK → references LightCones
- bid is a FK → references Builds

Stats(sid: INTEGER, stat_type: VARCHAR, stat_value: INTEGER, cid: INTEGER)

- (sid, cid) is the PK
- (cid, stat_type) is a CK
- cid is a FK → references Characters

Abilities (name: VARCHAR, ability_type: VARCHAR, ability_level: INTEGER, description: VARCHAR, cid: INTEGER)

- (name, cid) is the PK
- cid is a FK → references Characters

Basic(name: VARCHAR, cid: INTEGER, damage: INTEGER)

- (name, cid) is the PK

- name is a FK → references Abilities
- cid is a FK → references Characters

Skill(<u>name</u>: VARCHAR, <u>cid</u>: INTEGER, duration: INTEGER)

- (name, cid) is the PK
- name is a FK → references Abilities
- cid is a FK → references Characters

Ultimate(<u>name</u>: VARCHAR, <u>cid</u>: INTEGER, energy_cost: INTEGER)

- (name, cid) is the PK
- name is a FK → references Abilities
- cid is a FK → references Characters

Talents(<u>name:</u> VARCHAR, <u>cid</u>: INTEGER, effect: VARCHAR, trigger_condition: VARCHAR)

- (name, cid) is the PK
- name is a FK → references Abilities
- cid is a FK → references Characters

Consumables(<u>name</u>: VARCHAR, consumable_type: VARCHAR, effect: VARCHAR)

- name is the PK

Characters_Consumables(<u>name</u>: VARCHAR, <u>cid</u>: INTEGER)

- (name, cid) is the PK
- name is a FK → references Consumables
- cid is a FK → references Characters

LightCones(<u>cone_id</u>: INTEGER, name: VARCHAR, rarity: INTEGER, path: VARCHAR, effect: VARCHAR, bid: INTEGER)

- cone_id is the PK
- bid is a FK → references Builds

Materials(<u>mid</u>: INTEGER, name: VARCHAR, material_type: VARCHAR, location: VARCHAR, rarity: INTEGER)

- mid is the PK

Characters_Materials(<u>cid</u>: INTEGER, <u>mid</u>: INTEGER)

- (cid, mid) is the PK

- cid is a FK → references Characters
- mid is a FK → references Materials

Abilities_Materials(cid: INTEGER, name: VARCHAR, mid: INTEGER)

- (cid, name, mid) is the PK
- cid is a FK → references Characters
- name is a FK → references Abilities
- mid is a FK → references Materials

LightCones_Materials(cone_id: INTEGER, mid: INTEGER)

- (cone_id, mid) is the PK
- cone_id is a FK → references LightCones
- mid is a FK → references Materials

Relics_Materials(rid: INTEGER, mid: INTEGER)

- (rid, mid) is the PK
- rid is a FK → references Relics
- mid is a FK → references Materials

Builds(bid: INTEGER, name: VARCHAR)

- bid is the PK

Relics(rid: INTEGER, level: INTEGER, name: VARCHAR, main_stat: VARCHAR, rarity: INTEGER, relic_type: VARCHAR, set_name: VARCHAR, bid: INTEGER, rec_main: VARCHAR, rec_substat: VARCHAR)

- rid is the PK
- set_name is a FK (cannot be null) → references RelicSet
- bid is a FK → references Builds

RelicSet(name: VARCHAR, two_pb: VARCHAR, four_pb: VARCHAR)

- name is the PK

For naming purposes, attributes with similar names across entity sets have been renamed to have entity_attribute naming or first letter of entity + id for ids.

Functional Dependencies

Character

- cid → name, element, rarity, path,
- name → element, rarity, path
- cone_id → path

Stats

- sid, cid → stat_type, stat_value
- cid, stat_type → sid, stat_value

Abilities

- name, cid → ability_type, ability_level, description
- name → ability_type, description

Basic

- name, cid → damage

Skills

- name, cid → duration

Ultimates

- name, cid → energy_cost

Talent

- name, cid → trigger_condition, effect

Consumables

- name → type, effect

LightCones

- cid → name, rarity, path, effect
- name → effect, rarity, path

Materials

- mid → name, material_type, location, rarity
- name → material_type, location, rarity

Builds

- bid → name

Relics

- rid→ relic_level, name, main_stat, rarity, relic_type, set_name

- name → relic_type, set_name
- rid, bid → rec_main, rec_substat

RelicSet

- set_name → two_pb, four_pb

Character_Consumables

- No non-trivial dependencies

Characters_Materials

- No non-trivial dependencies

Abilities_Materials

- No non-trivial dependencies

LightCones_Materials

- No non-trivial dependencies

Relics_Materials

- No non-trivial dependencies

Normalization

Characters

1. cid → name, element, rarity, path
2. name → element, rarity, path
3. cone_id → path

- Steps:
  - Characters(<u>cid</u>, name, element, rarity, path, cone_id, bid)
  - Decompose by fd2:
    - R1(<u>name</u>, element, rarity, path)
    - R’(<u>cid</u>, name, cone_id, bid)
- Final Tables:
  - Characters(<u>name</u>, element, rarity, path)
  - CharacterRelations(<u>cid</u>, name, cone_id, bid)

Stats

1. sid, cid → stat_type, stat_value

- Stats(sid, stat_type, stat_value, cid)

Abilities

1. name, cid → ability_type, ability_level, description

- Abilities(name, ability_type, ability_level, cid, description,)

Basic

1. name, cid → damage

- Basic(name, cid, damage)

Skills

1. name, cid → duration

- Skill(name, cid, duration)

Ultimates

1. name, cid → energy_cost

- Ultimate(name, cid, energy_cost)

Talents

1. name, cid → trigger_condition, effect

- Talents(name, cid, trigger_condition, effect)

Consumables

1. name → type, effect

- Consumables(name, consumable_type, effect)

LightCones

1. cone_id → name, rarity, path, effect, bid
2. name → effect, rarity, path

- Steps:
  - LightCones(cone_id, name, rarity, path, effect, bid)
  - Decompose by fd2:
    - R1(cone_id, name, bid)
    - R’(name, rarity, path, effect)
- Final Tables:

- LightCones(cone_id, name, bid)
- LightConeDetails(name, rarity, path, effect)

Materials

1. mid → name, material_type, location, rarity
2. name → material_type, location, rarity

- Steps:
  - Materials(mid, name, material_type, location, rarity)
  - Decompose by fd2:
    - R1(name, material_type, location, rarity)
    - R’(mid, name)
- Final Tables:
  - Materials(mid, name)
  - MaterialDetails(name, material_type, location, rarity)

Builds

1. bid → name

- Builds(bid, name)

Relics

1. rid → relic_level, name, main_stat, rarity, relic_type, set_name, rec_main, rec_substat
2. name → relic_type, set_name

- Steps:
  - Relics(rid, relic_level, name, main_stat, rarity, relic_type, set_name, bid, rec_main, rec_substat)
  - Decompose by fd2:
    - R1(name, relic_type, set_name)
    - R’(rid, relic_level, name, main_stat, rarity, bid, rec_main, rec_substat)
- Final Tables:
  - Relics(rid, relic_level, name, main_stat, rarity, bid, rec_main, rec_substat)
  - RelicDetails(name, relic_type, set_name)

RelicSet

1. set_name → two_pb, four_pb

- RelicSet(set_name, two_pb, four_pb)

Characters_Consumables

- Characters_Consumables(name, cid)

Characters_Materials:

- Characters_Materials(cid, mid)

Abilities_Materials

- Abilities_Materials(cid, name, mid)

LightCones_Materials

- LightCones_Materials(cone_id, mid)

Relics_Materials

- Relics_Materials(rid, mid)

SQL DDL

```
CREATE TABLE Characters (
    name VARCHAR(100),
    element VARCHAR(100),
    rarity INTEGER,
    path VARCHAR(100),
    PRIMARY KEY (name),
)
```

```
CREATE TABLE Builds (
    bid INTEGER,
    name VARCHAR(100),
    PRIMARY KEY (bid)
)
```

```
CREATE TABLE LightConeDetails (
    name VARCHAR(100),
    rarity INTEGER,
    path VARCHAR(100),
    effect VARCHAR(500),
```

```
    PRIMARY KEY (name),
)
```

```
CREATE TABLE LightCones (
    cone_id INTEGER,
    name VARCHAR(100),
    bid INTEGER,
    PRIMARY KEY (cone_id),
    FOREIGN KEY (name) REFERENCES LightConeDetails
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (bid) REFERENCES Builds
)
```

- `ON DELETE CASCADE` added to foreign key `name`. If a Light Cone's details are removed, main reference should be removed.
- `ON UPDATE CASCADE` added to foreign key `name` to keep references to details updated appropriately.

```
CREATE TABLE CharacterRelations (
    cid INTEGER,
    name VARCHAR(100) NOT NULL,
    cone_id INTEGER,
    bid INTEGER,
    PRIMARY KEY (cid),
    FOREIGN KEY (name) REFERENCES Characters
        ON DELETE CASCADE,
    FOREIGN KEY (cone_id) REFERENCES LightCones
        ON DELETE SET NULL
        ON UPDATE CASCADE,
    FOREIGN KEY (bid) REFERENCES Builds
        ON DELETE SET NULL
        ON UPDATE CASCADE
)
```

- `ON DELETE CASCADE` added to foreign key `name`. If a character is removed from the database, all relations to it should also be removed.
- `ON DELETE SET NULL` added to foreign keys `cone_id` and `bid`. If a Light Cone or Build is deleted, the reference to them should be removed, but since Characters do not have total

participation, they should not be removed.

- `ON UPDATE CASCADE` added to foreign keys `cone_id` and `bid`. The corresponding reference to `LightCones` and `Builds` should be updated if it is ever changed.

```
CREATE TABLE Stats (
    sid INTEGER,
    stat_type VARCHAR(30),
    stat_value INTEGER,
    cid INTEGER,
    PRIMARY KEY (sid, cid),
    UNIQUE (cid, stat_type),
    FOREIGN KEY (cid) REFERENCES CharacterRelations
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

- `ON DELETE CASCADE` and `ON UPDATE CASCADE` added to foreign key `cid`. If a character is deleted, all of their corresponding stats should also be deleted. Reference to `CharacterRelations` should be updated if it is ever changed.

```
CREATE TABLE Abilities (
    name VARCHAR(100),
    ability_type VARCHAR(100),
    ability_level INTEGER,
    cid INTEGER,
    description VARCHAR(500),
    PRIMARY KEY (name, cid),
    FOREIGN KEY (cid) REFERENCES CharacterRelations
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

- `ON DELETE CASCADE` and `ON UPDATE CASCADE` added to foreign key `cid`. If a character is deleted, all of their corresponding abilities should also be deleted. Reference to `CharacterRelations` should be updated if it is ever changed.

```
CREATE TABLE Basic (
    name VARCHAR(100),
    cid INTEGER,
    damage VARCHAR(100),
```

```
    PRIMARY KEY (name, cid),
    FOREIGN KEY (name, cid) REFERENCES Abilities(name, cid)
        ON DELETE CASCADE
)

CREATE TABLE Skills (
    name VARCHAR(100),
    cid INTEGER,
    duration INTEGER,
    PRIMARY KEY (name, cid),
    FOREIGN KEY (name, cid) REFERENCES Abilities(name, cid)
        ON DELETE CASCADE
)

CREATE TABLE Ultimates (
    name VARCHAR(100),
    cid INTEGER,
    energy_cost INTEGER,
    PRIMARY KEY (name, cid),
    FOREIGN KEY (name, cid) REFERENCES Abilities(name, cid)
        ON DELETE CASCADE
)

CREATE TABLE Talents (
    name VARCHAR(100),
    cid INTEGER,
    trigger_condition VARCHAR(100),
    effect VARCHAR(200),
    PRIMARY KEY (name, cid),
    FOREIGN KEY (name, cid) REFERENCES Abilities(name, cid)
        ON DELETE CASCADE
```

- `ON DELETE CASCADE` was added to foreign key `(name, cid)` . If an ability is ever removed, its specific subclass should also be removed.

```
CREATE TABLE Consumables (
    name VARCHAR(100),
    consumable_type VARCHAR(100),
    effect VARCHAR(500),
```

```
    PRIMARY KEY (name)
)
```

```
CREATE TABLE MaterialDetails (
    name VARCHAR(100),
    material_type VARCHAR(100),
    location VARCHAR(100),
    rarity INTEGER,
    PRIMARY KEY (name)
)
```

```
CREATE TABLE Materials (
    mid INTEGER,
    name VARCHAR(100) NOT NULL,
    PRIMARY KEY (mid),
    FOREIGN KEY (name) REFERENCES MaterialDetails
        ON UPDATE CASCADE
        ON DELETE CASCADE
)
```

- `ON DELETE CASCADE` and `ON UPDATE CASCADE` added to foreign key `name`. If the details for a material is deleted, corresponding material should also be deleted. Likewise, any updates to `MaterialDetails` should update to `Materials` correspondingly.

```
CREATE TABLE RelicSet (
    set_name VARCHAR(100),
    two_pb VARCHAR(500),
    four_pb VARCHAR(500),
    PRIMARY KEY (set_name)
)
```

```
CREATE TABLE RelicDetails (
    name VARCHAR(100),
    relic_type VARCHAR(100),
    set_name VARCHAR(100) NOT NULL,
    PRIMARY KEY (name),
    FOREIGN KEY (set_name) REFERENCES RelicSet
        ON DELETE CASCADE
```

```
        ON UPDATE CASCADE
)
```

- `ON DELETE CASCADE` added to foreign key `set_name` . If a Relic set is deleted, all its corresponding relics should also be deleted.
- `ON UPDATE CASCADE` added to foreign keys `set_name`. The corresponding reference to `RelicSet` should be updated if it is ever changed.

```
CREATE TABLE Relics (
    rid INTEGER,
    relic_level INTEGER,
    name VARCHAR(100) NOT NULL,
    main_stat VARCHAR(30),
    rarity INTEGER,
    bid INTEGER,
    rec_main VARCHAR(100),
    rec_substat VARCHAR(100),
    PRIMARY KEY (rid),
    FOREIGN KEY (bid) REFERENCES Build
        ON DELETE SET NULL
        ON UPDATE CASCADE,
    FOREIGN KEY (name) REFERENCES RelicDetails
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

- `ON DELETE SET NULL` added to foreign key `bid`. If a build is deleted, the relic should stay the same but without an assigned build.
- `ON DELETE CASCADE` added to foreign key `name`. If the details for a relic are deleted, corresponding relic entry should also be deleted.
- `ON UPDATE CASCADE` added to both foreign keys. This is to keep the corresponding references of the keys updated.

```
CREATE TABLE Characters_Consumables (
    name VARCHAR(100),
    cid INTEGER,
    PRIMARY KEY (name, cid),
    FOREIGN KEY (name) REFERENCES Consumables
```

```
      ON DELETE CASCADE
      ON UPDATE CASCADE,
   FOREIGN KEY (cid) REFERENCES CharacterRelations
      ON DELETE CASCADE
      ON UPDATE CASCADE
)

CREATE TABLE Characters_Materials (
   cid INTEGER,
   mid INTEGER,
   PRIMARY KEY (cid, mid),
   FOREIGN KEY (cid) REFERENCES CharacterRelations
      ON DELETE CASCADE
      ON UPDATE CASCADE,
   FOREIGN KEY (mid) REFERENCES Materials
      ON DELETE CASCADE
      ON UPDATE CASCADE
)

CREATE TABLE Abilities_Materials (
   cid INTEGER,
   name VARCHAR(100),
   mid INTEGER,
   PRIMARY KEY (cid, name, mid)
   FOREIGN KEY (name, cid) REFERENCES Abilities(name, cid)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
   FOREIGN KEY (mid) REFERENCES Materials
      ON DELETE CASCADE
      ON UPDATE CASCADE
)

CREATE TABLE LightCones_Materials (
   cone_id INTEGER,
   mid INTEGER,
   PRIMARY KEY (cone_id, mid),
   FOREIGN KEY (cone_id) REFERENCES LightCones
      ON DELETE CASCADE
      ON UPDATE CASCADE,
   FOREIGN KEY (mid) REFERENCES Materials
```

```
        ON DELETE CASCADE
        ON UPDATE CASCADE
)

CREATE TABLE Relics_Materials (
    rid INTEGER,
    mid INTEGER,
    PRIMARY KEY (rid, mid),
    FOREIGN KEY (rid) REFERENCES Relics
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (mid) REFERENCES Materials
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

- `ON DELETE CASCADE` and `ON UPDATE CASCADE` has been added to all foreign keys in these relationship set tables. If one of the corresponding keys has been deleted, all related relations should also be deleted. If one of the corresponding keys has been updated, all related relations should also be updated.

Insert Examples

```
INSERT INTO Characters(name, element, rarity, path) VALUES
("Trailblazer", "Physical", 5, "The Destruction"),
("March 7th", "Ice", 4, "The Preservation"),
("Dan Heng", "Wind", 4, "The Hunt"),
("Himeko", "Fire", 5, "The Erudition"),
("Asta", "Fire", 4, "The Harmony")

INSERT INTO Builds(bid, name) VALUES
(0, "Destruction MC DPS"),
(1, "Himeko PF"),
(2, "Dan Heng Build"),
(3, "Himeko PF"),
(4, "March build")

INSERT INTO LightConeDetails(name, rarity, path, effect) VALUES
("Something Irreplaceable", 5, "The Destruction", "Kinship -- Increases the wea
rer's ATK by 24/28/32/36/40%. When the wearer defeats an enemy or is hit, im
```

mediately restores HP equal to 8/9/10/11/12% of the wearer's ATK. At the same t ime, the wearer's DMG is increased by 24/28/32/36/40% until the end of their next turn. This effect cannot stack and can only trigger 1 time per turn."),
("Moment of Victory", 5, "The Preservation", "Verdict -- Increases the wearer's DEF by 24/28/32/36/40% and Effect Hit Rate by 24/28/32/36/40%. Increases t he chance for the wearer to be attacked by enemies. When the wearer is attack ed, increase their DEF by an additional 24/28/32/36/40% until the end of the w earer's turn."),
("Swordplay", 4, "The Hunt", "Answers of Their Own -- For each time the weare r hits the same target, DMG dealt increases by 8/10/12/14/16%, stacking up to 5 time(s). This effect will be dispelled when the wearer changes targets."),
("Night on the Milky Way", 5, "The Erudition", "Meteor Swarm -- For every ene my on the field, increases the wearer's ATK by 9%/10.5%/12%/13.5%/15%, up t o 5 stacks. When an enemy is inflicted with Weakness Break, the DMG dealt by the wearer increases by 30%/35%/40%/45%/50% for 1 turn."),
("Meshing Cogs", 3, "The Harmony", "Fleet Triumph -- After the wearer uses att acks or gets hit, additionally regenerates 4/5/6/7/8 Energy. This effect can only be triggered 1 time per turn.")

INSERT INTO LightCones(cone_id, name, bid) VALUES
(0, "Something Irreplaceable", 0),
(1, "Moment of Victory", 1),
(2, "Swordplay", 2),
(3, "Night on the Milky Way", 3),
(4, "Meshing Cogs", 4)

INSERT INTO CharacterRelations(cid, name, cone_id, bid) VALUES
(0, "Trailblazer", NULL, 0),
(1, "March 7th", NULL, 1),
(2, "Dan Heng", NULL, 2),
(3, "Himeko", NULL, 3),
(4, "Asta", NULL, 4)

INSERT INTO Stats(sid, stat_type, stat_value, cid) VALUES
(10, "HP", 1058, 1),
(11, "Attack", 511, 1),
(12, "Defense", 573, 1),
(13, "Speed", 101, 1),
(14, "Taunt", 150, 1)

```
INSERT INTO Abilities(name, ability_type, ability_level, cid, description) VALUES
("Frigid Cold Arrow", "Single Target", 3, 1, "Deals Ice DMG."),
("The Power of Cuteness", "Defense", 5, 1, "Provides a single ally with a Shiel
d."),
("Glacial Cascade", "AoE ATK", 3, 1, "Deals Ice DMG to all enemies. Hit enemies
have a 50% base chance to be Frozen for 1 turn(s). While Frozen, the enemy ca
nnot take action and will receive additional Ice DMG equal to 30% of March 7t
h's ATK at the beginning of each turn."),
("Girl Power", "Single Target", 1, 1, "Deal Ice DMG after meeting condition. This
effect can be triggered 2 time(s) each turn."),
("Farewell Hit", "Single Target", 6, 0, "Deals Physical DMG."),
("RIP Home Run", "Blast", 9, 0, "Deals Physical DMG."),
("Stardust Ace", "Enhance", 7, 0, "Choose between two attack modes to deliver
full strike: Single Target/Blast"),
("Perfect Pickoff", "Enhance", 5, 0, "Enhance ATK. Can stack up to 2 time(s)."),
("Cloudlancer Art: North Wind", "Single Target", 3, 2, "Deals Wind DMG."),
("Cloudlancer Art: Torrent", "Single Target", 5, 2, "Deals Wind DMG. On CRIT hi
t, 100% base chance to reduce target's SPD by 12%."),
("Ethereal Dream", "Single Target", 5, 2, "Deals Wind DMG. If enemy is slowed,
DMG Multipler increases."),
("Superiority of Reach", "Enhance", 5, 2, "Enhance Wind RES PEN. Can be trigg
ered after 2 turn(s)."),
("Sawblade Tuning", "Single Target", 10, 3, "Deals Fire DMG."),
("Molten Detonation", "Blast", 10, 3, "Deals Fire DMG."),
("Heavenly Flare", "AoE", 10, 3, "Deals Fire DMG. Regenerates 5 extra Energy fo
r each enemy defeated."),
("Victory Rush", "AoE", 10, 3, "Consumes 3 charges to perform a follow-up atta
ck."),
("Spectrum Beam", "Single Target", 2, 4, "Deals Fire DMG"),
("Meteor Storm", "Bounce", 3, 4, "Deals Fire DMG. Further deals 4 extra times t
o random enemy."),
("Astral Blessing", "Support", 8, 4, "Increases SPD of all allies"),
("Astrometry", "Support", 9, 4, "Increases all allies' ATK for every stack of Char
ging")

INSERT INTO Basic(name, cid, damage) VALUES
("Frigid Cold Arrow", 1, "50-110% of ATK"),
("Farewell Hit", 0, "50-110% of ATK"),
("Cloudlancer Art: North Wind", 2, "50-110% of ATK"),
("Sawblade Tuning", 3, "50-110% ATK"),
```

```
("Spectrum Beam", 4, "50-110% of ATK")

INSERT INTO Skills(name, cid, duration) VALUES
("The Power of Cuteness", 1, 3),
("RIP Home Run", 0, 1),
("Cloudlancer Art: Torrent", 2, 2),
("Molten Detonation", 3, 1),
("Meteor Storm", 4, 1)

INSERT INTO Ultimates(name, cid, energy_cost) VALUES
("Glacial Cascade", 1, 120),
("Stardust Ace", 0, 120),
("Ethereal Dream", 2, 100),
("Heavenly Flare", 3, 120),
("Astral Blessing", 4, 120)

INSERT INTO Talents(name, cid, trigger_condition, effect) VALUES
("Girl Power", 1, "After shielded ally is attacked", "Counterattack"),
("Perfect Pickoff", 0, "Inflict Weakness Break on enemy", "ATK increases by 10-22%"),
("Superiority of Reach", 2, "Target of ally ability", "Wind RES PEN increases by 18-39.6%"),
("Victory Rush", 3, "Enemy inflicted with Weakness Break", "Gains 1 charge."),
("Astrometry", 4, "Enemy hit. Extra stack if enemy has Fire Weakness", "Gains 1 stack of Charging")
```

```
INSERT INTO Consumables(name, consumable_type, effect)  VALUES
("Alfalfa Salad", "Attack", "Increases all allies' CRIT Rate by 18% for the next battle."),
("All Good Potion", "Energy Regen", "Immediately regenerates 50% of Max Energy for a single ally."),
("Amber Huadiao Wine", "Defense", "Immediately causes all allies to lose HP equal to 5% of their Max HP, and increases their Max HP by 24% for the next battle."),
("Berrypheasant Skewers", "Restorative", "Immediately heals one ally by 15% of the ally's Max HP plus 150 extra HP."),
("Camo Paint", "Special", "Enemies will be less likely to detect your team for 75 s.")

INSERT INTO MaterialDetails(name, material_type, location, rarity) VALUES
```

("Destroyer's Final Road", "Trace", "Herta Space Station", 4),
("Endotherm Chitin", "Ascension", "Jarilo-VI", 4),
("Sparse Aether", "EXP", "All Locations", 2),
("Oath of Steel", "Ascension", "Herta Space Station", 3),
("Tracks of Destiny", "Trace", "Event rewards", 5),
("Lost Gold Fragment", "Relic", "Relic Salvage, Rewards", 3),
("Silvermane Badge", "Ascension", "Jarilo-VI", 2),
("Squirming Core", "Ascension, Trace", "Jarilo-VI", 4),
("Usurper's Scheme", "Ascension, Trace", "Herta Space Station", 3),
("Lifeless Blade", "Trace, Light Cone Ascension", "Herta Space Station", 3),
("Lost Crystal", "Relic", "Relic Salvage, Rewards", 4)

INSERT INTO Materials(mid, name) VALUES
(200, "Destroyer's Final Road"),
(30, "Endotherm Chitin"),
(0, "Sparse Aether"),
(11, "Oath of Steel"),
(100, "Tracks of Destiny"),
(5, "Lost Gold Fragment"),
(10, "Silvermane Badge"),
(6, "Squirming Core"),
(104, "Usurper's Scheme"),
(105, "Lifeless Blade"),
(106, "Lost Crystal")

INSERT INTO RelicSet(set_name, two_pb, four_pb) VALUES
("The Ashblazing Grand Duke", "Increases the DMG dealt by follow-up attacks by 20%.", "When the wearer uses follow-up attacks, increases the wearer's ATK by 6% for every time the follow-up attack deals DMG. This effect can stack up to 8 time(s) and lasts for 3 turn(s). This effect is removed the next time the wearer uses a follow-up attack."),
("Champion of Streetwise Boxing", "Increases Physical DMG by 10%.", "After the wearer attacks or is hit, their ATK increases by 5% for the rest of the battle. This effect can stack up to 5 time(s)."),
("Knight of Purity Palace", "Increases DEF by 15%.", "Increases the max DMG that can be absorbed by the Shield created by the wearer by 20%."),
("Sigonia, the Unclaimed Desolation", "Increases the wearer's CRIT Rate by 4%. When an enemy target gets defeated, the wearer's CRIT DMG increases by 4%, stacking up to 10 time(s).", NULL),
("Rutilant Arena", "Increases the wearer's CRIT Rate by 8%. When the wearer's

```
current CRIT Rate reaches 70% or higher, the wearer's Basic ATK and Skill DM
G increase by 20%.", NULL)

INSERT INTO RelicDetails(name, relic_type, set_name) VALUES
("Grand Duke's Crown of Netherflame", "Head", "The Ashblazing Grand Duke"),
("Champion's Chest Guard", "Body", "Champion of Streetwise Boxing"),
("Knight's Iron Boots of Order", "Feet", "Knight of Purity Palace"),
("Sigonia's Knot of Cyclicality", "Link Rope", "Sigonia, the Unclaimed Desolatio
n"),
("Taikiyan Laser Stadium", "Planar Sphere", "Rutilant Arena")

INSERT INTO Relics(rid, relic_level, name, main_stat, rarity, bid, rec_main, rec_s
ubstat) VALUES
(0, 10, "Grand Duke's Crown of Netherflame", "HP", 5, 1, "HP", "CRIT Rate, CRIT
DMG, ATK%"),
(1, 13, "Champion's Chest Guard", "CRIT DMG", 5, 0, "HP", "ATK%, CRIT Rate, C
RIT DMG, SPD"),
(2, 1, "Knight's Iron Boots of Order", "ATK%", 3, 4, "SPD, DEF%", "Effect Hit Rat
e, DEF%, SPD, HP%"),
(3, 14, "Sigonia's Knot of Cyclicality", "Energy Regen", 4, 1, "ATK%", "CRIT Rate,
CRIT DMG, SPD"),
(4, 12, "Taikiyan Laser Stadium", "HP%", 5, 2, "ATK%", "CRIT Rate, CRIT DMG,
ATK%, SPD")
```

```
INSERT INTO Characters_Consumables(name, cid) VALUES
("Alfalfa Salad", 0),
("Amber Huadiao Wine", 1),
("Alfalfa Salad", 2),
("Camo Paint", 3),
("All Good Potion", 3)

INSERT INTO Characters_Materials(cid, mid) VALUES
(0, 104),
(1, 11),
(2, 6),
(3, 30),
(4, 30)

INSERT INTO Abilities_Materials(cid, name, mid) VALUES
(0, "Farewell Hit", 105),
```

```
(1, "Glacial Cascade", 100),
(2, "Superiority of Reach", 200),
(3, "Heavenly Flare", 200),
(4, "Astral Blessing", 100)


INSERT INTO LightCones_Materials(cone_id, mid) VALUES
(1, 11),
(1, 10),
(0, 105),
(2, 0),
(3, 0)

INSERT INTO Relics_Materials(rid, mid) VALUES
(0, 5),
(1, 5),
(1, 106),
(2, 5),
(3, 106),
(4, 5)
```

AI Usage Acknowledgment

We did not use any AI.