

# CPSC 304 Project Cover Page

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

Date: \_\_\_\_Mar 1<sup>th</sup>, 2024\_\_\_\_

Group Number: \_\_\_\_26\_\_\_\_

| Name        | Student Number | CS Alias (Userid) | Preferred E-mail Address |
|-------------|----------------|-------------------|--------------------------|
| Emma Huang  | 79107488       | w4h2q             | ehuang07@student.ubc.ca  |
| Maggie Weng | 74030461       | m7w9w             | yweng07@student.ubc.ca   |
| Oscar Yik   | 98217094       | e1u9w             | oyik@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

1. Cover page
2. Our project allows users to create and keep track of their own topics, notes, quizzes, and flashcards to help them study. They can also share their study material with other users, search for other user's study material, and join study groups. Users can also search for and use other user's shared study notes.
3. ER Diagram: At the end of the PDF
4. Schema

PKs are underlined, FKs are **bolded**, comments begin with //

### **Achievement**

Achievement(title: string, image: string, difficulty: string, points: integer, borderColor: string)

Not Null: image, difficulty, points, borderColor

FOREIGN KEY (difficulty) REFERENCES AchievementLevel

### **Obtains**

Obtains(**username**: string, **achievementTitle**: string)

FOREIGN KEY (username) REFERENCES User

FOREIGN KEY (achievementTitle) REFERENCES Achievement(title)

### **User**

User(username: string, password: string, email: string, points: integer, school: string, schoolLogo: string, schoolLogo, reputation: string, borderColor: string)

Candidate Key: email

Unique: email

Not Null: reputation, borderColor, password, email, points

### **CreatesTopic**

CreatesTopic(Id: string, **username**: string, title: string, privacyInfo: string, description: string, lastOpened: Date)

Not Null: title, privacyInfo, lastOpened

FOREIGN KEY (username) REFERENCES User

### **Tag**

Tag(name: string, color: string)

Not Null: color

### **Has**

Has(**tagName**: string, **topicId**: string)

FOREIGN KEY (tagName) REFERENCES Tag(name)

FOREIGN KEY (topicId) REFERENCES CreatesTopic(id)

### **ContainsStudyMaterial**

ContainsStudyMaterial(title: string, **topicId**: string, privacyInfo: string, description: string, lastOpened: Date, dateCreated: Date, parsedText: string, highScore: integer, type: string, icon: string)

Not Null: privacyInfo, lastOpened, Type, Icon

FOREIGN KEY (topicId) REFERENCES CreatesTopic(id)

### **OwnsQuizQuestion**

OwnsQuizQuestion(id: string, **studyMatTitle**: string, **topicId**: string, question: string, answer: string type: string, difficulty: string, points: integer)

Not Null: question, answer, type, difficulty, points

FOREIGN KEY (studyMatTitle, topicId) REFERENCES ContainsStudyMaterial (title, topicId)

### **OwnsCard**

OwnsCard(id: string, **studyMatTitle**: string, **topicId**: string, question: string, answer: string, image: string)

Not Null: question, answer

FOREIGN KEY (studyMatTitle, topicId) REFERENCES ContainsStudyMaterial (title, topicId)

### **Group**

Group(code: string, name: string)

Not Null: name

### **Joins**

Joins(username: string, **groupCode**: string)

FOREIGN KEY (username) REFERENCES User

FOREIGN KEY (groupCode) REFERENCES Group(code)

### **Shares**

Shares(**studyMaterialTitle**: string, **topicId**: string, **groupCode**: string)

FOREIGN KEY (studyMaterialTitle, topicId) REFERENCES ContainsStudyMaterial(title, topicId)

FOREIGN KEY (groupCode) REFERENCES Group(code)

### **Likes**

Likes(**studyMaterialTitle**: string, **topicId**: string, username: string)

FOREIGN KEY (studyMaterialTitle, topicId) REFERENCES ContainsStudyMaterial(title, topicId)

FOREIGN KEY (username) REFERENCES User

## **5. Functional Dependencies**

### **Achievement:**

title → image, difficulty, points, borderColor (PK)

difficulty → points, borderColor

**User:**

username → password, email, points, school, schoolLogo, reputation, borderColor (PK)

email → username, password, points, school, schoolLogo, reputation, borderColor (CK)

school → schoolLogo

reputation → borderColor

**CreatesTopic:**

id → title, privacy info, description, username (PK)

**Tag:**

name → color (PK)

**ContainsStudyMaterial:**

title, topicId → privacyInfo, description, lastOpened, dateCreated, parsedText, highScore, type, icon (PK)

type → icon

**OwnsQuizQuestion:**

id → studyMatTitle, topicId, question, answer, difficulty, points, type (PK)

type → difficulty

difficulty → points

**OwnsCard:**

id → studyMatTitle, topicId, question, answer, image (PK)

**Group:**

code → name (PK)

**Relations with no FDs:** Obtains, Has, Joins, Shares, Likes

## 6. Normalization

We normalize everything to BCNF.

**Achievement: =====**

title → image, difficulty, points, borderColor (PK)

difficulty → points, borderColor // violates BCNF

Decompose on difficulty → points, borderColor:

**Normalized Relations:**

**Achievement**(title, difficulty, image)

**AchievementLevel**(difficulty, points, borderColor)

**User:** =====

username → password, email, points, school, schoolLogo, reputation, borderColor (PK)

email → username, password, points, school, schoolLogo, reputation, borderColor (CK)

school → schoolLogo // violates BCNF

reputation → borderColor // violates BCNF

Decompose on school → schoolLogo

**R1**(username, **school**, password, email, points, reputation, borderColor) // reputation → borderColor violates BCNF

**R2**(school, schoolLogo)

Decompose again on R1 reputation → borderColor:

**R3**(username, **school**, **reputation**, password, email, points)

**R4**(reputation, borderColor)

**Normalized Relations:**

**School**(name, schoolLogo) **We renamed the “school” attribute in School to “name”**

**User**(username, **school**, **reputation**, password, email, points)

**Reputation**(reputation, borderColor)

**ContainsStudyMaterial:** =====

title, topicId → privacyInfo, description, lastOpened, dateCreated, parsedText, highScore, type, icon (PK)

type → icon // violates BCNF

Decompose on type → icon:

**Normalized Relations:**

**ContainsStudyMaterial**(title, topicId, **type**, privacyInfo, description, lastOpened, dateCreated, parsedText, highScore)

**StudyMaterialType**(type, icon)

**OwnsQuizQuestion:** =====

id → studyMatTitle, topicId, question, answer, difficulty, points, type (PK)

type → difficulty // violates BCNF

difficulty → points // violates BCNF

Decompose on type → difficulty:

**Normalized Relations:**

**OwnsQuizQuestion**(id, **studyMatTitle**, **topicId**, **type**, question, answer, points)

**QuizQuestionDifficulty**(type, difficulty)

## **All Tables after Normalization:**

### **Achievement:**

Achievement(title: string, **difficulty**: string, image: string)

Not Null: difficulty, image

FOREIGN KEY (difficulty) REFERENCES AchievementLevel

### **AchievementLevel**

AchievementLevel(difficulty: string, points: integer, borderColor: string)

Not Null: points, borderColor

### **Obtains**

Obtains(username: string, **achievementTitle**: string)

FOREIGN KEY (username) REFERENCES User

FOREIGN KEY (achievementTitle) REFERENCES Achievement(title)

### **User**

User(username: string, **school**: string, **reputation**: string, password: string, email: string, points: integer)

Candidate keys: email

Unique: email

Not Null: password, reputation, email, points

FOREIGN KEY (school) REFERENCES School(name)

FOREIGN KEY (reputation) REFERENCES Reputation

### **School**

School(name: string, schoolLogo: string)

Not Null: schoolLogo

### **Reputation**

Reputation(reputation: string, borderColor: string)

Not Null: borderColor

### **CreatesTopic**

CreatesTopic(id: string, **username**: string, title: string, privacyInfo: string, description: string, lastOpened: Date)

Not Null: title, privacyInfo, lastOpened

FOREIGN KEY (username) REFERENCES User

### **Tag**

Tag(name: string, color: string)

Not Null: color

## Has

Has(**tagName**: string, **topicId**: string)

FOREIGN KEY (tagName) REFERENCES Tag(name)

FOREIGN KEY (topicId) REFERENCES CreatesTopic(id)

## ContainsStudyMaterial

ContainsStudyMaterial(**title**: string, **topicId**: string, **type**: string, privacyInfo: string, description: string, lastOpened: Date, dateCreated: Date, parsedText: string, highScore: integer)

Not Null: privacyInfo, lastOpened, type

FOREIGN KEY (topicId) REFERENCES CreatesTopic(id)

FOREIGN KEY (type) REFERENCES StudyMaterialType

## StudyMaterialType

StudyMaterialType(**type**: string, icon: string)

Not Null: icon

## OwnsQuizQuestion

OwnsQuizQuestion(**id**: string, **studyMatTitle**: string, **topicId**: string, **type**: string, question: string, answer: string, points: integer)

Not Null: question, answer, type, points

FOREIGN KEY (studyMatTitle, topicId) REFERENCES ContainsStudyMaterial (title, topicId)

FOREIGN KEY (type) REFERENCES QuizQuestionDifficulty

## QuizQuestionDifficulty

QuizQuestionDifficulty(**type**: string, difficulty: string)

Not Null: difficulty

## OwnsCard

OwnsCard(**id**: string, **studyMatTitle**: string, **topicId**: string, question: string, answer: string, image: string)

Not Null: question, answer

FOREIGN KEY (studyMatTitle, topicId) REFERENCES ContainsStudyMaterial (title, topicId)

## Group

Group(**code**: string, name: string)

Not Null: name

## Joins

Joins(**username**: string, **groupCode**: string)

FOREIGN KEY (username) REFERENCES User

FOREIGN KEY (groupCode) REFERENCES Group(code)

## Shares

Shares(**studyMaterialTitle**: string, **topicId**: string, **groupCode**: string)

FOREIGN KEY (studyMaterialTitle, topicId) REFERENCES ContainsStudyMaterial(title, topicId)  
FOREIGN KEY (groupId) REFERENCES Group(code)

### Likes

Likes(**studyMaterialTitle**: string, **topicId**: string, **username**: string)

FOREIGN KEY (studyMaterialTitle, topicId) REFERENCES ContainsStudyMaterial(title, topicId)

FOREIGN KEY (username) REFERENCES User

## 7. SQL DDL

Bolded tables are the ones that have been affected by normalization in Q6.

```
CREATE TABLE Achievement (  
    title          VARCHAR          PRIMARY KEY,  
    difficulty     VARCHAR          NOT NULL  
    image         VARCHAR          NOT NULL,  
    FOREIGN KEY (difficulty) REFERENCES AchievementLevel  
)
```

```
CREATE TABLE AchievementLevel (  
    difficulty     VARCHAR          PRIMARY KEY,  
    points        INT              NOT NULL,  
    borderColor   VARCHAR          NOT NULL  
)
```

```
CREATE TABLE Obtains (  
    username      VARCHAR,  
    achievementTitle VARCHAR,  
    PRIMARY KEY (username, achievementTitle),  
    FOREIGN KEY (username) REFERENCES User ON DELETE CASCADE,  
    FOREIGN KEY (achievementTitle) REFERENCES Achievement(title) ON  
DELETE CASCADE  
)
```

```
CREATE TABLE User (  
    username      VARCHAR PRIMARY KEY,  
    school        VARCHAR,  
    reputation    VARCHAR NOT NULL,  
    password      VARCHAR NOT NULL,  
    email         VARCHAR UNIQUE NOT NULL,  
    points        INT NOT NULL,
```



```

        FOREIGN KEY (school) REFERENCES School(name) ON DELETE SET
NULL,
        FOREIGN KEY (reputation) REFERENCES Reputation
)

```

```

CREATE TABLE School (
    name          VARCHAR PRIMARY KEY,
    schoolLogo    VARCHAR NOT NULL
)

```

```

CREATE TABLE Reputation (
    reputation     VARCHAR PRIMARY KEY,
    borderColor    VARCHAR NOT NULL
)

```

```

CREATE TABLE CreatesTopic (
    id             VARCHAR PRIMARY KEY,
    username       VARCHAR,
    title          VARCHAR NOT NULL,
    privacyInfo    VARCHAR NOT NULL,
    description     VARCHAR,
    lastOpened     DATE NOT NULL,
    FOREIGN KEY (username) REFERENCES User ON DELETE CASCADE
)

```

```

CREATE TABLE Tag (
    name           VARCHAR PRIMARY KEY,
    color          VARCHAR NOT NULL
)

```

```

CREATE TABLE Has (
    tagName        VARCHAR,
    topicId        VARCHAR,
    PRIMARY KEY (tagName, topicId),
    FOREIGN KEY (tagName) REFERENCES Tag(name) ON DELETE CASCADE,
    FOREIGN KEY (topicId) REFERENCES CreatesTopic(id) ON DELETE
CASCADE
)

```

```

CREATE TABLE ContainsStudyMaterial(
    title          VARCHAR,
    topicId        VARCHAR,
    type           VARCHAR NOT NULL,
    privacyInfo    VARCHAR NOT NULL,

```

```

        lastOpened          DATE NOT NULL,
        dateCreated          DATE NOT NULL,
        description          VARCHAR,
        parsedText           VARCHAR,
        highScore            INT,
        PRIMARY KEY (title, topicId),
        FOREIGN KEY (topicId) REFERENCES CreatesTopic(id) ON DELETE
CASCADE
        FOREIGN KEY (type) REFERENCES StudyMaterialType
)

CREATE TABLE StudyMaterialType(
    type          VARCHAR PRIMARY KEY,
    icon          VARCHAR NOT NULL,
)

CREATE TABLE OwnsQuizQuestion (
    id            VARCHAR PRIMARY KEY,
    studyMatTitle VARCHAR,
    topicId       VARCHAR,
    type          VARCHAR NOT NULL,
    question      VARCHAR NOT NULL,
    answer        VARCHAR NOT NULL,
    points        INT,
    FOREIGN KEY (studyMatTitle, topicId) REFERENCES
ContainsStudyMaterial (title, topicId) ON DELETE CASCADE,
    FOREIGN KEY (type) REFERNECES QuizQuestionDifficulty(type)
)

CREATE TABLE QuizQuestionDifficulty (
    type          VARCHAR PRIMARY KEY,
    icon          VARCHAR NOT NULL
)

CREATE TABLE OwnsCard (
    id            VARCHAR PRIMARY KEY,
    studyMatTitle VARCHAR,
    topicId       VARCHAR,
    question      VARCHAR NOT NULL,
    answer        VARCHAR NOT NULL,
    image         VARCHAR,
    FOREIGN KEY (studyMatTitle, topicId) REFERENCES ContainsStudyMaterial
(title, topicId) ON DELETE CASCADE
)

```

```

CREATE TABLE Group (
    code          CHAR[10] PRIMARY KEY,
    name          VARCHAR NOT NULL
)

```

```

CREATE TABLE Joins(
    username          VARCHAR,
    groupCode         CHAR[10],
    PRIMARY KEY (username, groupCode),
    FOREIGN KEY (username) REFERENCES User ON DELETE CASCADE,
    FOREIGN KEY (groupCode) REFERENCES Group(code) ON DELETE
CASCADE
)

```

```

CREATE TABLE Shares(
    studyMaterialTitle  VARCHAR,
    topicId             VARCHAR
    groupCode           CHAR[10],
    PRIMARY KEY (studyMaterialTitle, topicId, groupCode),
    FOREIGN KEY (studyMaterialTitle, topicId) REFERENCES
ContainsStudyMaterial(title, topicId) ON DELETE CASCADE,
    FOREIGN KEY (groupCode) REFERENCES Group(code) ON DELETE
CASCADE
)

```

```

CREATE TABLE Likes(
    studyMaterialTitle  VARCHAR,
    topicId             VARCHAR
    username            VARCHAR,
    PRIMARY KEY (studyMaterialTitle, topicId, groupCode),
    FOREIGN KEY (studyMaterialTitle, topicId) REFERENCES
ContainsStudyMaterial(title, topicId) ON DELETE CASCADE,
    FOREIGN KEY (username) REFERENCES User ON DELETE CASCADE,
)

```

## 8. INSERT statements

INSERT

```
INSERT INTO Achievement (title, difficulty, image)
VALUES ('First Topic!', 'Easy' , 'https://t.ly/FvkQO")
```

```
INSERT INTO Achievement (title, difficulty, image)
VALUES ('First Quiz!', 'Easy' , 'https://t.ly/FvkQT")
```

```
INSERT INTO Achievement (title, difficulty, image)
VALUES ('First Flash Card Set!', 'Easy' , 'https://t.ly/FvkQU")
```

```
INSERT INTO Achievement (title, difficulty, image)
VALUES ('You Are the Best!', 'Impossible' , 'https://t.ly/FvkQI")
```

```
INSERT INTO Achievement (title, difficulty, image)
VALUES ('Learned Everything', 'Impossible' , 'https://t.ly/FvkQK")
```

```
INSERT INTO AchievementLevel (difficulty, points, borderColor)
VALUES ('Easy', '5', 'green')
```

```
INSERT INTO AchievementLevel (difficulty, points, borderColor)
VALUES ('Medium', '10', 'yellow')
```

```
INSERT INTO AchievementLevel (difficulty, points, borderColor)
VALUES ('Hard', '20', 'red')
```

```
INSERT INTO AchievementLevel (difficulty, points, borderColor)
VALUES ('Very Hard', '50', 'Blue')
```

```
INSERT INTO AchievementLevel (difficulty, points, borderColor)
VALUES ('Impossible', '100000000000', 'black')
```

```
INSERT INTO Obtains (username, achievementTitle)
VALUES ('i_am_a_beginner', 'First Topic!')
```

```
INSERT
```

```
INSERT INTO Obtains (username, achievementTitle)
VALUES ('i_am_a_beginner', 'First Quiz!')
```

```
INSERT INTO Obtains (username, achievementTitle)
VALUES ('i_am_a_beginner', 'First Flash Card Set!')
```

```
INSERT INTO Obtains (username, achievementTitle)
VALUES ('smug_person', 'You Are the Best!')
```

```
INSERT INTO Obtains (username, achievementTitle)
VALUES ('hacker101', 'Learned Everything')
```

```
INSERT INTO User (username, school, reputation, password, email, points)
VALUES ('hacker101', 'UBC', '999999', 'SHig32b8wug', 'hacker@gmail.com', z)
```

```
INSERT INTO User (username, school, reputation, password, email, points)
VALUES ('i_am_a_beginner', 'UBC', '8', '1234', 'myemail@gmail.com', a)
```

```
INSERT INTO User (username, school, reputation, password, email, points)
VALUES ('smug_person', 'UBC', '696969', 'lmao', 'best@gmail.com', g)
```

```
INSERT INTO User (username, school, reputation, password, email, points)
VALUES ('admin', 'UBCO', '0', 'h21s$7', 'admin@hotmail.net', z)
```

```
INSERT INTO User (username, school, reputation, password, email, points)
VALUES ('ditto07', 'Uva Academy', '100', 'iwannabetheverybest', 'abc@uva.com', c)
```

```
INSERT INTO School (name, schoolLogo)
VALUES ('UBC', https://t.ly/F21s)
```

```
INSERT INTO School (name, schoolLogo)
VALUES ('UBCO', https://t.ly/F1s)
```

```
INSERT
```

```
INTO School (name, schoolLogo)
VALUES ('Uva Academy', 'https://t.ly/F81s)
```

```
INSERT
INTO School (name, schoolLogo)
VALUES ('SFU', 'https://t.ly/G21s)
```

```
INSERT
INTO School (name, schoolLogo)
VALUES ('KPU', 'https://t.ly/A21s)
```

```
INSERT
INTO Reputation (reputation, borderColor)
VALUES ('a', 'green')
```

```
INSERT
INTO Reputation (reputation, borderColor)
VALUES ('c', 'yellow')
```

```
INSERT
INTO Reputation (reputation, borderColor)
VALUES ('f', 'orange')
```

```
INSERT
INTO Reputation (reputation, borderColor)
VALUES ('g', 'blue')
```

```
INSERT
INTO Reputation (reputation, borderColor)
VALUES ('z', 'black')
```

```
INSERT
INTO Topic (id, username, title, privacyInfo, description, lastOpened)
VALUES ('1', 'i_am_a_beginner', 'times_tables', 'private', 'multiplication practice for
grade four', 2024-03-01)
```

```
INSERT
INTO Topic (id, username, title, privacyInfo, description, lastOpened)
VALUES ('2', 'smug_person', 'Calculus 9', 'public', 'Secrets of the Universe', 2021-12-25)
```

```
INSERT
INTO Topic (id, username, title, privacyInfo, description, lastOpened)
VALUES ('3', 'smug_person', 'List of Personal Achievements', 'public', 'I am so cool',
2024-03-01)
```

```
INSERT
INTO Topic (id, username, title, privacyInfo, description, lastOpened)
VALUES ('4', 'admin', 'nuclear_codes', 'classified', 'post to internet when app is dead',
2002-02-02)
```

```
INSERT
INTO Topic (id, username, title, privacyInfo, description, lastOpened)
VALUES ('5', 'ditto07', 'gen6pokemon', 'public', 'memorize gen 6 pokemon before
pokemon z-a comes out', 2024-03-01)
```

```
INSERT
INTO Tag (name, color)
VALUES ('Math', 'red')
```

```
INSERT
INTO Tag (name, color)
VALUES ('Games', 'green')
```

```
INSERT
INTO Tag (name, color)
VALUES ('Top', 'blue')
```

```
INSERT
INTO Tag (name, color)
VALUES ('UBC', 'yellow')
```

```
INSERT
INTO Tag (name, color)
VALUES ('DO NOT READ', 'black')
```

```
INSERT
INTO Has (tagName, topicId)
VALUES ('Math', 'times_tables')
```

```
INSERT
INTO Has (tagName, topicId)
VALUES ('Math', 'Calculus 9')
```

```
INSERT
INTO Has (tagName, topicId)
VALUES ('Games', 'gen6pokemon')
```

```
INSERT
```

```
INTO Has (tagName, topicId)
VALUES ('Games', 'List Of Personal Achievements')
```

```
INSERT
INTO Has (tagName, topicId)
VALUES ('DO NOT READ', 'nuclear_codes')
```

```
INSERT
INTO ContainsStudyMaterial (title, topicId, type, privacyInfo, lastOpened, dateCreated,
description, parsedText, highScore)
VALUES ('waterTypes', 'gen6pokemon', 'Flashcard Set', 'public', 2024-02-29,
2024-02-28, 'Flashcards for memorizing gen6 water type pokemon', NULL, NULL)
```

```
INSERT
INTO ContainsStudyMaterial (title, topicId, type, privacyInfo, lastOpened, dateCreated,
description, parsedText, highScore)
VALUES ('steelTypes', 'gen6pokemon', 'Flashcard Set', 'public', 2024-02-29,
2024-02-28, 'Flashcards for memorizing gen6 steel type pokemon', NULL, NULL)
```

```
INSERT
INTO ContainsStudyMaterial (title, topicId, type, privacyInfo, lastOpened, dateCreated,
description, parsedText, highScore)
VALUES ('7_timestables', 'times_tables', 'Quiz', 'public', 2024-03-01, 2024-03-01,
'SUPER HARD', NULL, 0.3)
```

```
INSERT
INTO ContainsStudyMaterial (title, topicId, type, privacyInfo, lastOpened, dateCreated,
description, parsedText, highScore)
VALUES ('Quantum Computing Quadruple Integral Wave ... idk some other bs',
'Calculus 9', 'Flashcard Set', 'public', 2021-12-25, 2021-08-26, 'ONLY FOR SUPER
SMART PEOPLE', NULL, NULL)
```

```
INSERT
INTO ContainsStudyMaterial (title, topicId, type, privacyInfo, lastOpened, dateCreated,
description, parsedText, highScore)
VALUES ('Big missile', 'nuclear_codes', 'Quiz', 'classified', 2024-03-01, 1999-12-31, 'I
need to pass or else our missiles will go off', NULL, 1)
```

```
INSERT
INTO StudyMaterialType (type, icon)
VALUES ('Flashcard Set', 'https://t.ly/2i1ihqria1s)
```

```
INSERT
INTO StudyMaterialType (type, icon)
```



VALUES ('Quiz', '<https://t.ly/2iriq htigwria1s>')

INSERT

INTO StudyMaterialType (type, icon)

VALUES ('Parsed Notes', '<https://t.ly/2iwquhr2iqa1s>')

INSERT

INTO StudyMaterialType (type, icon)

VALUES ('Super Flashcard Set', '<https://t.ly/2ir2uq r2a1s>')

INSERT

INTO StudyMaterialType (type, icon)

VALUES ('MEGA QUIZ', 'https://t.ly/2i1i21wuia1s')

INSERT

INTO OwnsQuizQuestion (id, studyMatTitle, topicId, type, question, answer, points)

VALUES ('1', '7\_timestables', 'times\_tables', 'easy', '1', '7', 0.1)

INSERT

INTO OwnsQuizQuestion (id, studyMatTitle, topicId, type, question, answer, points)

VALUES ('2', '7\_timestables', 'times\_tables', 'medium', '4', '28', 0.1)

INSERT

INTO OwnsQuizQuestion (id, studyMatTitle, topicId, type, question, answer, points)

VALUES ('3', '7\_timestables', 'times\_tables', 'hard', '7', '49', 0.1)

INSERT

INTO OwnsQuizQuestion (id, studyMatTitle, topicId, type, question, answer, points)

VALUES ('4', 'Big missile', 'nuclear\_codes', 'impossible', 'Neo-Armstrong Missile',  
'81husv17qUl1#\*H21', 0.5)

INSERT

INTO OwnsQuizQuestion (id, studyMatTitle, topicId, type, question, answer, points)

VALUES ('5', 'Big missile', 'nuclear\_codes', 'impossible', 'doom',  
'ALwqkngNW(!&%( @32', 0.5)

INSERT

INTO QuizQuestionDifficulty (type, icon)

VALUES ('easy', '<https://t.ly/2ir2uq r2a1s>')

INSERT

INTO QuizQuestionDifficulty (type, icon)

VALUES ('medium', '<https://t.ly/2r2a1s>')

```
INSERT
INTO QuizQuestionDifficulty (type, icon)
VALUES ('hard', 'https://t.ly/2ir2ukqq\_r2a1s)
```

```
INSERT
INTO QuizQuestionDifficulty (type, icon)
VALUES ('very hard', 'https://t.ly/2921q\_r2a1s)
```

```
INSERT
INTO QuizQuestionDifficulty (type, icon)
VALUES ('impossible', 'https://t.ly/2ir1s)
```

```
INSERT
INTO OwnsCard (id, studyMatTitle, topicId, question, answer, image)
VALUES ('1', 'waterTypes', 'gen6pokemon', 'emo ninja', 'Greninja', 'https://t.ly/2ir2uqr2a)
```

```
INSERT
INTO OwnsCard (id, studyMatTitle, topicId, question, answer, image)
VALUES ('2', 'waterTypes', 'gen6pokemon', 'gun for arm', 'Clawitzer',
'https://t.ly/2ir2uqqkw\_r2a)
```

```
INSERT
INTO OwnsCard (id, studyMatTitle, topicId, question, answer, image)
VALUES ('3', 'steelTypes', 'gen6pokemon', 'double swords', 'Doublade',
'https://t.ly/2ir2uq819j)
```

```
INSERT
INTO OwnsCard (id, studyMatTitle, topicId, question, answer, image)
VALUES ('4', 'steelTypes', 'gen6pokemon', 'keys', 'Klefki', 'https://t.ly/q\_r2a)
```

```
INSERT
INTO OwnsCard (id, studyMatTitle, topicId, question, answer, image)
VALUES ('5', 'Quantum Computing Quadruple Integral Wave ... idk some other bs',
'Calculus 9', 'Riemann Hypothesis?', 'true', 'https://t.ly/2ir2u91hjjq\_r2a)
```

```
INSERT
INTO Group (code, name)
VALUES ('1', 'cool_kidz')
```

```
INSERT
INTO Group (code, name)
VALUES ('2', 'even_cooler_kidzz')
```

```
INSERT  
INTO Group (code, name)  
VALUES ('3', 'pokemon fans')
```

```
INSERT  
INTO Group (code, name)  
VALUES ('4', 'UBC CS')
```

```
INSERT  
INTO Group (code, name)  
VALUES ('5', 'DO NOT JOIN')
```

```
INSERT  
INTO Joins (username, groupCode)  
VALUES ('i_am_a_beginner', '1')
```

```
INSERT  
INTO Joins (username, groupCode)  
VALUES ('smug_person', '1')
```

```
INSERT  
INTO Joins (username, groupCode)  
VALUES ('sumg_person', '2')
```

```
INSERT  
INTO Joins (username, groupCode)  
VALUES ('smug_person', '5')
```

```
INSERT  
INTO Joins (username, groupCode)  
VALUES ('ditto07', '3')
```

```
INSERT  
INTO Shares (studyMaterialTitle, topicId, groupCode)  
VALUES ('7_timestable', 'times_tables', 1)
```

```
INSERT  
INTO Shares (studyMaterialTitle, topicId, groupCode)  
VALUES ('waterTypes', 'gen6pokemon', 3)
```

```
INSERT  
INTO Shares (studyMaterialTitle, topicId, groupCode)  
VALUES ('steelTypes', 'gen6pokemon', 3)
```

```
INSERT
INTO Shares (studyMaterialTitle, topicId, groupCode)
VALUES ('Quantum Computing Quadruple Integral Wave ... idk some other bs',
'Calculus 9', 1)
```

```
INSERT
INTO Shares (studyMaterialTitle, topicId, groupCode)
VALUES ('Quantum Computing Quadruple Integral Wave ... idk some other bs',
'Calculus 9', 3)
```

```
INSERT
INTO Likes (studyMaterialTitle, topicId, username)
VALUES ('Quantum Computing Quadruple Integral Wave ... idk some other bs',
'Calculus 9', 'i_am_a_beginner')
```

```
INSERT
INTO Likes (studyMaterialTitle, topicId, username)
VALUES ('waterTypes', 'gen6pokemon', 'ditto07')
```

```
INSERT
INTO Likes (studyMaterialTitle, topicId, username)
VALUES ('steelTypes', 'gen6pokemon', 'ditto07')
```

```
INSERT
INTO Likes (studyMaterialTitle, topicId, username)
VALUES('Quantum Computing Quadruple Integral Wave ... idk some other bs', 'Calculus
9', 'smug_person')
```

```
INSERT
INTO Likes (studyMaterialTitle, topicId, username)
VALUES('Quantum Computing Quadruple Integral Wave ... idk some other bs', 'Calculus
9', 'admin')
```

#### ER Diagram changes:

- Removed User Produces StudyMaterial relationship
  - This can be modeled in code logic and would not serve a significant purpose in the database model
- Removed Weak-Entity to ISA relationship
  - As suggested by TA over email
- Refactored Quiz and FlashcardSet entity into StudyMaterial with different types as suggested by our TA over email
- Removed “Generates” and “makes” relationships while refactoring the weak entity to ISA
- Added various attributes to create meaningful non-PK or CK functional dependencies:
  - User: school, schoolLogo, reputation, borderColor
  - Achievement: difficulty, points, borderColor
  - StudyMaterial: type, icon
  - QuizQuestion: type, difficulty, points
- Added other attributes that we felt would be useful for our app:
  - Topic: lastOpened (we want to sort by last opened date)
  - StudyMaterial: lastOpened
  - Card: image (we want users to have the options to include an image on their flashcards)
  - User: points (these are the total points a user has earned from achievements)
- Added StudyMaterialComponent as a parent entity of both QuizQuestion and Card
  - This reduced the total redundancy of the model as both QuizQuestion and Card had  $\frac{3}{4}$  of their attributes overlap previously
  - This fulfills the ISA requirement from milestone 1
  - Also included the Disjoint + Total specifications on the ISA that was missing previously
- Added “Owns” relationship between new entity StudyMaterialComponent and StudyMaterial to reproduce the previous “Quiz has QuizQuestions” and “FlashCardSet has Card” relationship

