University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: October 20, 2023

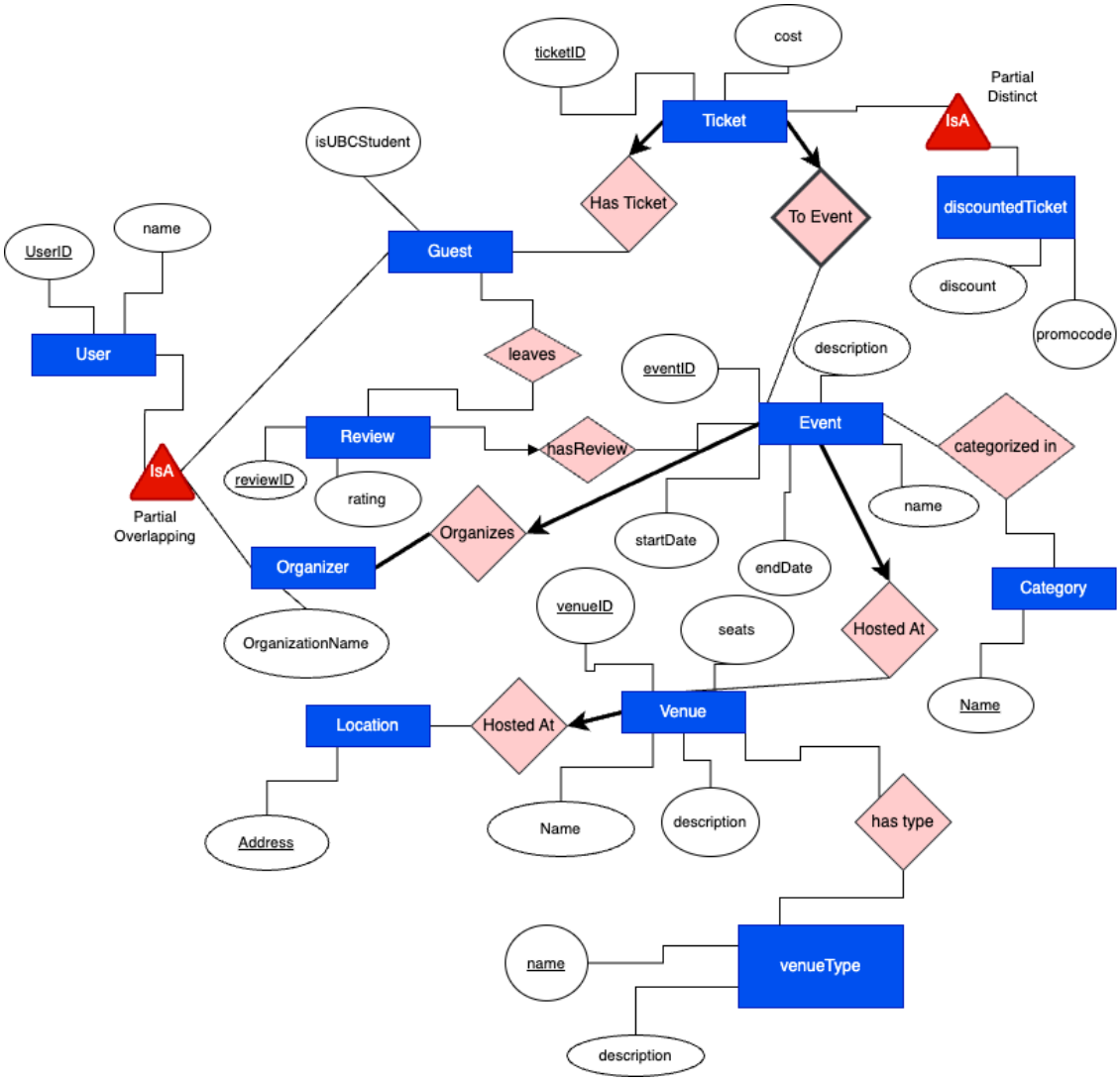
Group Number: 150

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
asar	75185298	H3u8k	asar@asarmichil.com
arden	12549432	C3a2t	general@ardensinclair.com
crystal	99260986	s5x5a	parvenucrystal@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
Event Calendar:
Our app would work as a platform within the events and entertainment industry, allowing organizers to host events with details, categories, and tickets, while attendees / guests can search for events, buy tickets, and leave reviews.
The ER diagram
Updates: Event and category is now a many to many relationship, added constraints to ticket IsA, and fields to Guest and Organizer.



University of British Columbia, Vancouver
Department of Computer Science

Schema

Underline: Primary Key

Review(ReviewID INT, rating INT, guestID)

VenueType(name STR, description STR)

Category(name STR, description STR)

User(UserID INT, name STR)

Organizer(UserID INT FK, organizationName STR)

Guest(UserID INT FK, isUBCStudent BOOLEAN)

Ticket(TicketID INT, cost INT, guestID INT FK, eventID INT FK)

Discounted Tickets(TicketID FK, Discount INT, promocode INT)

Event(EventID INT, description STR, startDate INT, endDate INT, name STR, venue INT FK, Organizer INT FK, category STR FK)

EventInCategory(EventID FK, CategoryName FK)

Location(postalCode STR, city STR, province STR, country STR, streetNumber INT, streetSecondLine STR, streetName, STR)

Venue(VenueID INT, Name STR, description STR, seats INT, venueType)

VenueType(name STR, description STR)

Functional Dependencies

UserID → name
VenueID →name, description, seats, location
eventID → name, description, startDate, endDate, venueID, organizerID
reviewID → rating
ticketID → cost
postalCode, country → city, province
Province → country
country, province, city, streetName, streetNumber, → postalCode
categoryName → categoryDescription

University of British Columbia, Vancouver
Department of Computer Science

Normalisation

Review(ReviewID INT, rating INT) is in BCNF

VenueType(name STR, description STR) is in BCNF

Category(name STR, description STR) is in BCNF

User(UserID INT, name STR) is in BCNF

Ticket(TicketID INT, COST INT) is in BCNF

Discounted Tickets(TicketID FK, Discount INT, promocode INT)

Event(EventID INT, description STR, startDate INT, endDate INT, name STR, venue INT FK, Organizer INT FK, category STR FK) is in BCNF

EventInCategory(EventID FK, CategoryName FK) is in BCNF

Organizer(UserID INT FK, organizationName STR) is in BCNF

Venue(VenueID INT, Name STR, description STR, seats INT, venueType) is in BCNF

VenueType(name STR, description STR) is in BCNF

postalCode, country → city, province violates 3NF, so Location is not in 3NF need to decompose:
R1(Streetnumber, Streetsecondline, Streetname, Postalcode, country) is in BCNF
R2(City, Province, postalcode, country)

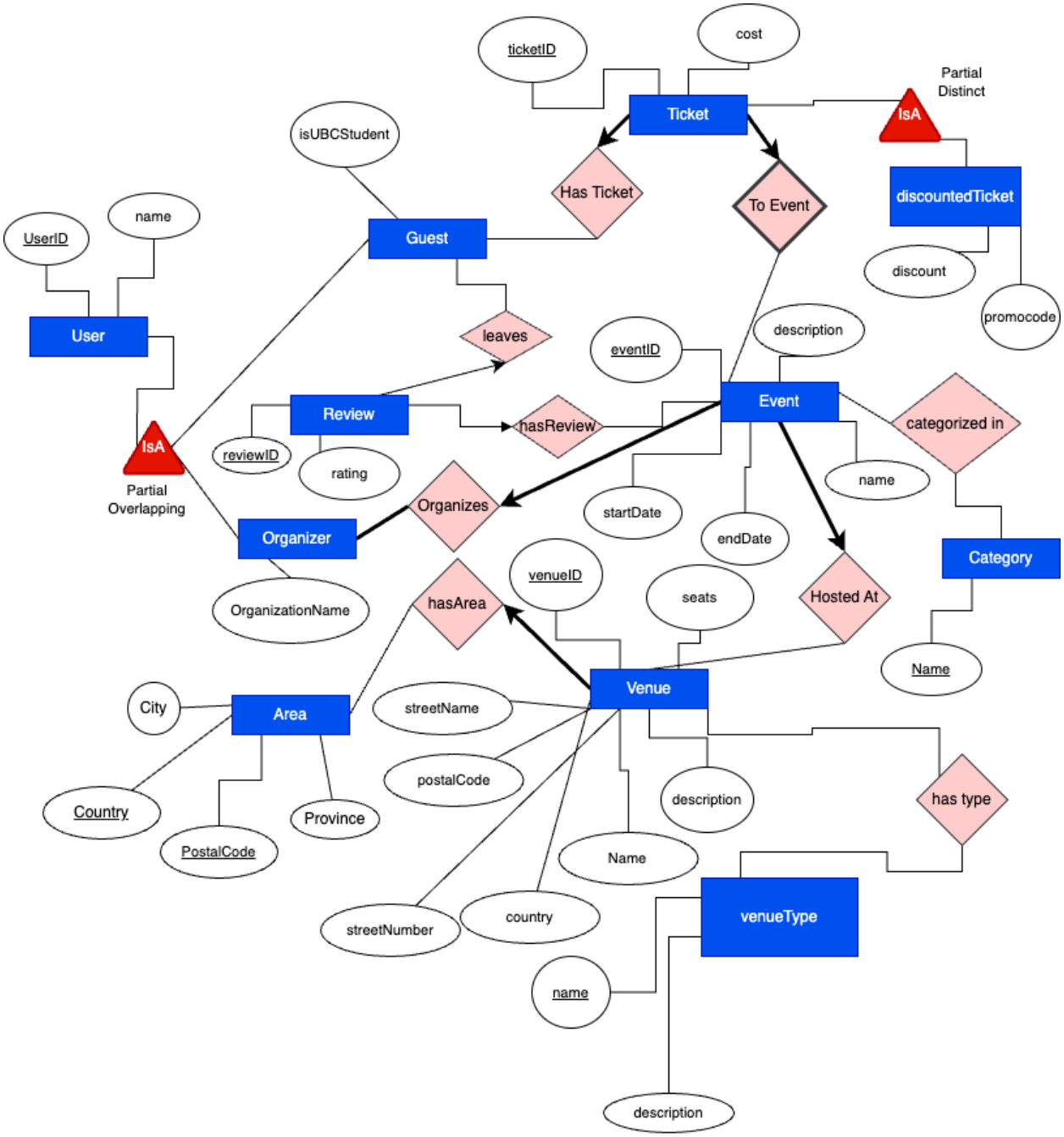
Province → country violates BCNF so will decompose R2:
R3(Postalcode, City, Province)
R4(Province, City) is in BCNF

postalCode, country → city, province violates BCNF so decompose R3:
r5(Country, Postalcode) is in BCNF
r6(City, Province) is in BCNF

Decompositions: R1(Streetnumber, Streetsecondline, Streetname, Postalcode, country), R4(Province, City), r5(Country, Postalcode)

After Normalisation New Diagram:

Our database needed to be updated with Venue now including fields StreetName, postalcode, country, streetnumber and a reference to area which has postal code, country city and province. We also dropped the location table.
The new diagram would be:



SQL statements:

```
CREATE TABLE users (
	user_id SERIAL PRIMARY KEY NOT NULL,
	first_name TEXT NOT NULL,
	last_name TEXT NOT NULL
);

CREATE TABLE organizer (
	user_id SERIAL PRIMARY KEY NOT NULL
		REFERENCES users(user_id)
		ON UPDATE CASCADE
		ON DELETE CASCADE,
	organization_name TEXT
);

CREATE TABLE guest (
	user_id SERIAL PRIMARY KEY NOT NULL
		REFERENCES users(user_id)
		ON UPDATE CASCADE
		ON DELETE CASCADE,
	is_ubc_student BOOLEAN NOT NULL
);

CREATE TABLE venue_type (
	venue_type_name TEXT PRIMARY KEY NOT NULL,
	description TEXT NOT NULL
);

CREATE TABLE area (
	postal_code CHAR(6) NOT NULL,
	country TEXT NOT NULL,
	city TEXT NOT NULL,
	province TEXT NOT NULL,
	PRIMARY KEY (postal_code, country)
);

CREATE TABLE venue (
	venue_id SERIAL PRIMARY KEY NOT NULL,
```

```
	name TEXT NOT NULL,
	description TEXT NOT NULL,
	seats INTEGER,
	venue_type_name TEXT NOT NULL
		REFERENCES venue_type(venue_type_name)
		ON UPDATE CASCADE
		ON DELETE RESTRICT,
	postal_code CHAR(6) NOT NULL,
	country TEXT NOT NULL,
	street_number INTEGER NOT NULL,
	street_name TEXT NOT NULL,
	FOREIGN KEY (postal_code, country)
		REFERENCES area(postal_code, country)
		ON UPDATE RESTRICT
		ON DELETE RESTRICT
);

CREATE TABLE event (
	event_id SERIAL PRIMARY KEY NOT NULL,
	name TEXT NOT NULL,
	description TEXT NOT NULL,
	start_date DATE NOT NULL,
	end_date DATE NOT NULL,
	organizer_id SERIAL NOT NULL
		REFERENCES organizer(user_id)
		ON UPDATE CASCADE
		ON DELETE CASCADE,
	venue_id SERIAL NOT NULL
		REFERENCES venue(venue_id)
		ON UPDATE CASCADE
		ON DELETE CASCADE
);

CREATE TABLE category (
	category_name TEXT PRIMARY KEY NOT NULL,
	description TEXT NOT NULL
);

CREATE TABLE event_in_category (
	event_id SERIAL NOT NULL
```

```
                REFERENCES event(event_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        category_name TEXT NOT NULL
                REFERENCES category(category_name)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        PRIMARY KEY (event_id, category_name)
);

CREATE TABLE ticket (
        ticket_id SERIAL NOT NULL,
        event_id SERIAL NOT NULL
                REFERENCES event(event_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        user_id SERIAL
                REFERENCES guest(user_id)
                ON UPDATE CASCADE
                ON DELETE SET NULL,
        cost NUMERIC(20) NOT NULL,
        PRIMARY KEY (ticket_id, event_id)
);

CREATE TABLE discounted_ticket (
        ticket_id SERIAL NOT NULL,
        event_id SERIAL NOT NULL,
        discount NUMERIC(3) NOT NULL,
        promo_code TEXT NOT NULL,
        PRIMARY KEY (ticket_id, event_id),
        FOREIGN KEY (ticket_id, event_id)
                REFERENCES ticket(ticket_id, event_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE review (
        review_id SERIAL PRIMARY KEY NOT NULL,
        rating NUMERIC(2) NOT NULL,
        comment VARCHAR(512),
```

```
        user_id SERIAL NOT NULL
                REFERENCES guest(user_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        event_id SERIAL NOT NULL
                REFERENCES event(event_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);
```

Insertions:

```
INSERT INTO
      users (first_name, last_name)
VALUES
      ('John', 'Doe'),
      ('John', 'Bro'),
      ('Michael', 'Jackson'),
      ('Asar', 'Zed'),
      ('Arden', 'Regson');

INSERT INTO
      organizer (user_id, organization_name)
VALUES
      (1, 'AMS Events'),
      (2, 'XYZ Events'),
      (3, 'Happy Events'),
      (4, 'Best Events'),
      (5, 'CVC Events');

INSERT INTO
      guest (user_id, is_ubc_student)
VALUES
      (1, false),
      (2, true),
      (3, false),
      (4, true),
      (5, false);

INSERT INTO
      venue_type (venue_type_name, description)
VALUES
      (
      'Concert Hall',
      'Large indoor venue for music concerts'
      ),
      (
      'Conference Center',
      'Space for conferences and seminars'
      ),
      ('Stadium', 'Large sports and entertainment venue'),
```

```
      ('Theater', 'Venue for stage performances'),
      ('Outdoor', 'Open-air space for events');

INSERT INTO
      area (postal_code, country, city, province)
VALUES
      ('V6T1Z4', 'Canada', 'Vancouver', 'BC'),
      ('M6P2L4', 'Canada', 'Toronto', 'ON'),
      ('90210', 'USA', 'Beverly Hills', 'CA'),
      ('M5S2K5', 'Canada', 'Toronto', 'ON'),
      ('M5G2C4', 'Canada', 'Toronto', 'ON');

INSERT INTO
      venue (
      name,
      description,
      seats,
      venue_type_name,
      postal_code,
      country,
      street_number,
      street_name
      )
VALUES
      (
      'Grand Arena',
      'Large concert hall',
      5000,
      'Concert Hall',
      'V6T1Z4',
      'Canada',
      123,
      'Main Street'
      ),
      (
      'Convention Center',
      'Modern conference space',
      1000,
      'Conference Center',
      'M6P2L4',
```

```
      'Canada',
      60,
      'Mountview Avenue'
      ),
      (
      'Sports Stadium',
      'Football and sports events',
      30000,
      'Stadium',
      '90210',
      'USA',
      567,
      'Sports Road'
      ),
      (
      'Downtown Theater',
      'Intimate theater experience',
      500,
      'Theater',
      'M5S2K5',
      'Canada',
      101,
      'Theater Lane'
      ),
      (
      'UBC Field',
      'Outdoor field for events',
      NULL,
      'Outdoor',
      'M5G2C4',
      'Canada',
      200,
      'Park Avenue'
      );

INSERT INTO
      event (
      name,
      description,
      start_date,
```

```
    end_date,
    organizer_id,
    venue_id
    )
    VALUES
    (
    'Hozier Concert',
    'A night of folk music',
    '2023-11-15',
    '2023-11-16',
    1,
    1
    ),
    (
    'Angular Tech Conference',
    'Innovations in Angular Web Framework',
    '2023-10-25',
    '2023-10-27',
    2,
    2
    ),
    (
    'UBC Mens Ultimate Vs Waterloo',
    'The big game',
    '2023-12-01',
    '2023-12-01',
    3,
    3
    ),
    (
    'UBC Theatre Hamlet',
    'Drama and comedy',
    '2023-11-20',
    '2023-11-21',
    4,
    4
    ),
    (
    'UBC CVC picnic in the Park',
    'Outdoor student event',
```

```
      '2023-09-05',
      '2023-09-05',
      5,
      5
      );

INSERT INTO
      category (category_name, description)
VALUES
      ('Music', 'Events related to music and concerts'),
(
      'Technology',
      'Tech-related events and conferences'
      ),
      ('Sports', 'Sports and athletic events'),
      ('Theater', 'Stage performances and plays'),
      ('Outdoor', 'Open-air and outdoor events');

INSERT INTO
      event_in_category (event_id, category_name)
VALUES
      (1, 'Music'),
      (2, 'Technology'),
      (3, 'Sports'),
      (4, 'Theater'),
      (5, 'Outdoor'),
      (5, 'Music');

INSERT INTO
      ticket (event_id, user_id, cost)
VALUES
      (1, 2, 4000),
      (2, 4, 15000),
      (3, 3, 7500),
      (4, 5, 5500),
      (5, 3, 1000);

INSERT INTO
      discounted_ticket (ticket_id, event_id, discount, promo_code)
VALUES
```

```
      (1, 1, 10, 'ROCK10'),
      (2, 2, 20, 'TECH20'),
      (3, 3, 15, 'SPORTS15'),
      (4, 4, 5, 'THEATER5'),
      (5, 5, 2, 'BACK2SCHOOL');

INSERT INTO
      review (rating, comment, user_id, event_id)
VALUES
      (4, 'Great concert!', 2, 1),
      (5, 'I Love Angular So Much!', 4, 2),
      (4, 'GO UBC!', 3, 3),
      (
      4,
      'All time favourite production of Hamlet!',
      5,
      4
      ),
      (2, 'It rained :(.', 1, 5);
```