Cisc 332 Project - Part 3

Adam Perron, Brandon Bloch, Shannon Klett

Sections

Section 1: All Assumptions

Application Design

- There will be very few administrative users, and at least one will have direct access to the database. Therefore we do not need a front end feature for requesting, adding, or removing admin privileges. If there is a new admin user, they can be upgraded directly in the database.
- Suppliers should NOT be allowed to add reviews on their own properties.
- Suppliers should be allowed to make bookings on their own properties.
- Users can either search by all districts or one, all types or one, all features or one, maximum price or any combination of these.

Overall Database Design

- We chose to give lots of extra space for varchars.
  - We're assuming our database won't be big enough for this excess to matter.
  - Better safe than sorry!
- We're using tables for constant string values, such as booking status type, property type, faculty, and degree type.
  - This will help keep data consistent in our system, and allow us to easily add new types.
- NULL values are generally not allowed unless we have good reason. Columns containing strings will default to an empty string, which, unlike NULL, will not cause type errors when processing in frontend.
- Delete and cancel action actually remove the record from the database, rather than using a "deleted" column.

Bookings

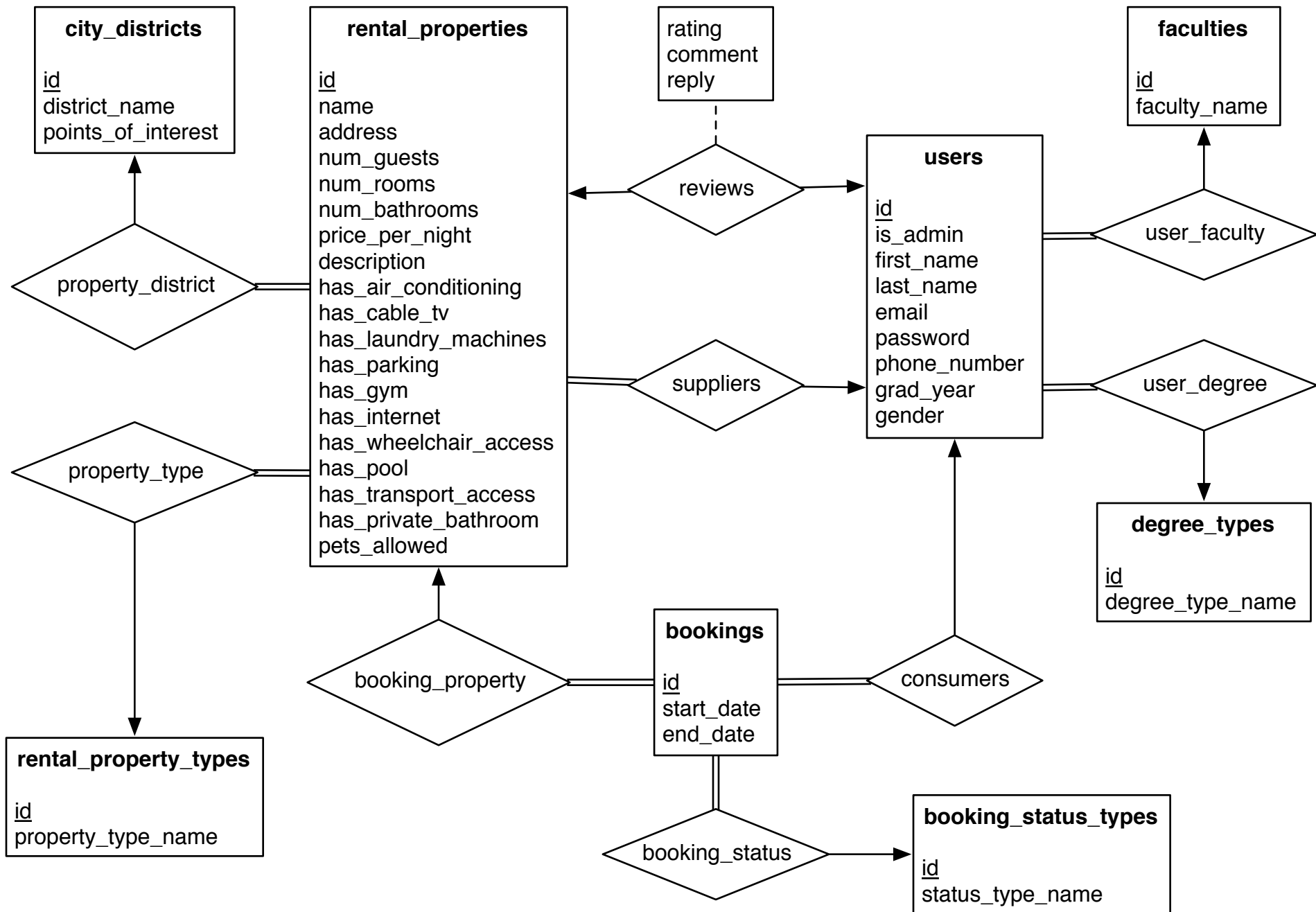- Bookings have their own id to make them easily searchable.

Rentals

- All properties must be in exactly one district. Districts do not overlap. If the property is on the border of districts, supplier must select one.
  - The same applies to property type and supplier; every property must have exactly one type and supplier.
- price_per_night is in whole Canadian dollars (no cent values are allowed), and represents the price for one day/night (ie noon day 1 to 11am day 2). Total price for a booking will be this value multiplied by the number of days.
- Features are listed as individual boolean columns (which are actually type tinyint), to make them easily searchable. Columns can be easily added and the default value is false.

Reviews

- A user can leave at most one comment/rating on a property, and the supplier can only reply once.
- When giving a review, users must give a rating, but the comment is optional.

Users

- Longest phone number should be 15 digits, plus 6 digits for extension , and extra space just in case.
- grad_year is the 4 digit number representing the year they graduated (ie 2016). If user has multiple degrees from Queen's we'll use the most recent year.
- Gender is a varchar field to allow for non-binary genders. Users can identify as they wish.

city_districts
- <u>id</u>
- district_name
- points_of_interest

rental_properties
- <u>id</u>
- name
- address
- num_guests
- num_rooms
- num_bathrooms
- price_per_night
- description
- has_air_conditioning
- has_cable_tv
- has_laundry_machines
- has_parking
- has_gym
- has_internet
- has_wheelchair_access
- has_pool
- has_transport_access
- has_private_bathroom
- pets_allowed

rating
comment
reply

reviews

users
- <u>id</u>
- is_admin
- first_name
- last_name
- email
- password
- phone_number
- grad_year
- gender

faculties
- <u>id</u>
- faculty_name

user_faculty

user_degree

degree_types
- <u>id</u>
- degree_type_name

property_district

property_type

rental_property_types
- <u>id</u>
- property_type_name

suppliers

booking_property

bookings
- <u>id</u>
- start_date
- end_date

consumers

booking_status

booking_status_types
- <u>id</u>
- status_type_name

Section 3: Relational Schema

```sql
CREATE TABLE `booking_status_types` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `status_type_name` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`id`)
)

CREATE TABLE `bookings` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `consumer_id` int(11) unsigned NOT NULL,
  `property_id` int(11) unsigned NOT NULL,
  `start_date` date NOT NULL,
  `end_date` date NOT NULL,
  `status_id` int(2) unsigned NOT NULL,
  PRIMARY KEY (`id`),
  KEY `consumer_id` (`consumer_id`),
  KEY `property_id` (`property_id`),
  KEY `status_id` (`status_id`),
  CONSTRAINT `bookings_ibfk_1` FOREIGN KEY (`consumer_id`) REFERENCES `users` (`id`),
  CONSTRAINT `bookings_ibfk_2` FOREIGN KEY (`property_id`) REFERENCES `rental_properties` (`id`),
  CONSTRAINT `bookings_ibfk_3` FOREIGN KEY (`status_id`) REFERENCES `booking_status_types` (`id`)
)

CREATE TABLE `city_districts` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `district_name` varchar(255) NOT NULL DEFAULT '',
  `points_of_interest` text NOT NULL,
  PRIMARY KEY (`id`)
)

CREATE TABLE `degree_types` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `degree_type_name` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`id`)
)

CREATE TABLE `faculties` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `faculty_name` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`id`)
)

CREATE TABLE `rental_properties` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(255) NOT NULL,
  `supplier_id` int(11) unsigned NOT NULL,
```

```sql
  `address` varchar(255) NOT NULL,
  `district_id` int(11) unsigned NOT NULL,
  `property_type_id` int(2) unsigned NOT NULL,
  `num_guests` int(10) unsigned NOT NULL DEFAULT '0',
  `num_rooms` int(10) unsigned NOT NULL DEFAULT '0',
  `num_bathrooms` int(10) unsigned NOT NULL,
  `price` int(5) NOT NULL,
  `description` text NOT NULL,
  `has_air_conditioning` tinyint(11) NOT NULL DEFAULT '0',
  `has_cable_tv` tinyint(11) NOT NULL DEFAULT '0',
  `has_laundry_machines` tinyint(11) NOT NULL DEFAULT '0',
  `has_parking` tinyint(11) NOT NULL DEFAULT '0',
  `has_gym` tinyint(11) NOT NULL DEFAULT '0',
  `has_internet` tinyint(11) NOT NULL DEFAULT '0',
  `pets_allowed` tinyint(11) NOT NULL DEFAULT '0',
  `has_wheelchair_access` tinyint(11) NOT NULL DEFAULT '0',
  `has_pool` tinyint(11) NOT NULL DEFAULT '0',
  `has_transport_access` tinyint(11) NOT NULL DEFAULT '0',
  `has_private_bathroom` tinyint(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
  KEY `district_id` (`district_id`),
  KEY `supplier_id` (`supplier_id`),
  KEY `property_type_id` (`property_type_id`),
  CONSTRAINT `rental_properties_ibfk_1` FOREIGN KEY (`district_id`) REFERENCES `city_districts` (`id`),
  CONSTRAINT `rental_properties_ibfk_2` FOREIGN KEY (`supplier_id`) REFERENCES `users` (`id`),
  CONSTRAINT `rental_properties_ibfk_3` FOREIGN KEY (`property_type_id`) REFERENCES
`rental_property_types` (`id`)
)

CREATE TABLE `rental_property_types` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `property_type_name` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`id`)
)

CREATE TABLE `reviews` (
  `consumer_id` int(11) unsigned NOT NULL,
  `property_id` int(11) unsigned NOT NULL,
  `rating` int(1) unsigned NOT NULL,
  `comment` text NOT NULL,
  `reply` text NOT NULL,
  PRIMARY KEY (`consumer_id`,`property_id`),
  KEY `property_id` (`property_id`),
  CONSTRAINT `reviews_ibfk_1` FOREIGN KEY (`consumer_id`) REFERENCES `users` (`id`),
  CONSTRAINT `reviews_ibfk_2` FOREIGN KEY (`property_id`) REFERENCES `rental_properties` (`id`)
)
```

```sql
CREATE TABLE `users` (
 `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
 `is_admin` tinyint(1) NOT NULL DEFAULT '0',
 `first_name` varchar(50) NOT NULL,
 `last_name` varchar(50) NOT NULL,
 `email` varchar(100) NOT NULL,
 `password` varchar(255) NOT NULL,
 `phone_number` varchar(30) NOT NULL,
 `grad_year` int(4) NOT NULL,
 `faculty_id` int(2) unsigned NOT NULL,
 `degree_type_id` int(2) unsigned NOT NULL,
 `gender` varchar(255) NOT NULL,
 PRIMARY KEY (`id`),
 KEY `faculty_id` (`faculty_id`),
 KEY `degree_type_id` (`degree_type_id`),
 CONSTRAINT `users_ibfk_1` FOREIGN KEY (`faculty_id`) REFERENCES `faculties` (`id`),
 CONSTRAINT `users_ibfk_2` FOREIGN KEY (`degree_type_id`) REFERENCES `degree_types` (`id`)
)
```

logged in?

no → 1. Welcome/Login

yes → 3. Main Page

1. Welcome/Login

click "Register" → 2. Registration

click "Cancel" ← 2. Registration

1.1 submit form → success?

success? — no → 1. Welcome/Login

success? — yes → 3. Main Page

2. Registration

2.1 submit form → success?

success? — no → 2. Registration

success? — yes → 3. Main Page

11.2 click "Cancel Membership" → 2. Registration

3. Main Page

3.1 as consumer, cancel a pending/confirmed booking
3.2 as supplier, accept a pending booking
3.3 as a supplier, reject a pending booking
3.4 as a supplier, cancel a confirmed booking

click "My Properties" → 7. My Properties

click "Search" → 4. Search

click "Settings" → 11. Settings

click "Admin" → 12. Administrator

click a listing → 5. Property Details

7. My Properties

click "Add" → 8. Add Property

8.1 submit form → success?

success? — yes → 7. My Properties

success? — no → 8. Add Property

click "Edit" on a listing → 9. Edit Property

9.1 submit form → success?

success? — yes → 7. My Properties

success? — no → 9. Edit Property

click "Delete" on a listing → 10. Delete Property

redirect → 7. My Properties

9. Edit Property — click "Delete" → 10. Delete Property

4. Search

4.1 submit search

click a listing → 5. Property Details

5. Property Details

5.1 submit a review
5.2 submit a reply
5.3 delete a review
5.4 delete a reply

5.5 as an admin, delete the property → 12. Administrator

click start date of desired booking → 6. Booking Request

11. Settings

11.1 submit profile form

12. Administrator

12.1 delete a member

6. Booking Request

6.1 submit request → success?

success? — yes → 5. Property Details

success? — no (show an error) → 6. Booking Request

click "Cancel" → 5. Property Details