University of British Columbia, Department of Computer Science

CPSC 304

Cover Page for Project Design

Date: October 14, 2018

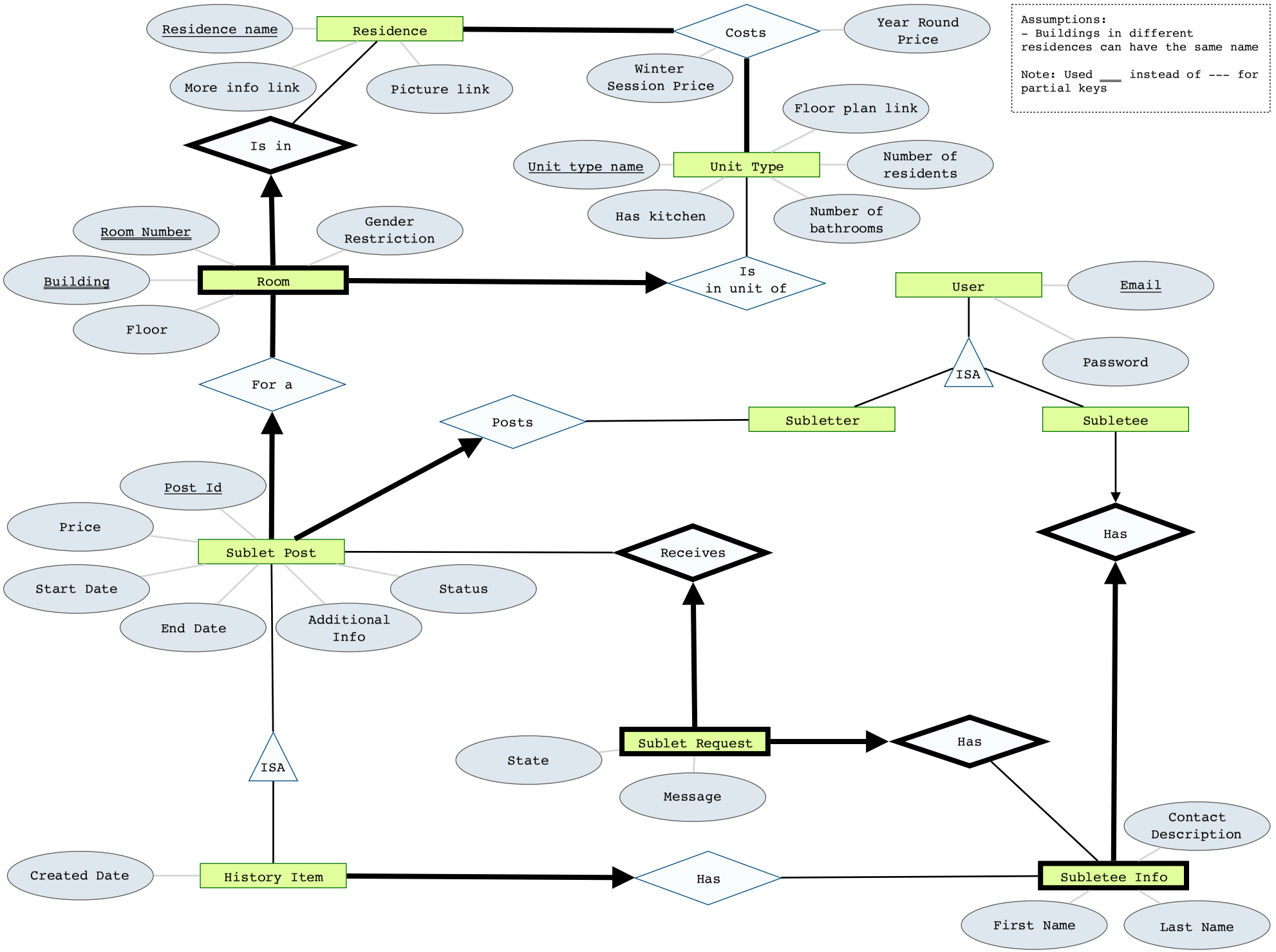
Group Members:

Name	Student Number	CS Userid	Tutorial Section	Email Address
Edmund Oh	17335167	d4b1b	T1H	1edmundoh@gmail.com
Shamit Rahman	19965152	x2l0b	T1G	rahmanshamit@gmail.com
Guilherme Lameira de Almeida	20318151	k3n0b	T1E	gui.l.a@hotmail.com
Rodolfo Orozco	10282168	t3x0b	T1C	raov97@outlook.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 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

ER Diagram


Residence name
Residence
Costs
Year Round Price
Winter Session Price
More info link
Picture link
Is in
Floor plan link
Unit type name
Unit Type
Number of residents
Has kitchen
Number of bathrooms
Room Number
Gender Restriction
Building
Room
Is in unit of
Floor
For a
User
Email
Password
ISA
Posts
Subletter
Subletee
Post Id
Price
Sublet Post
Receives
Has
Start Date
Status
End Date
Additional Info
ISA
State
Sublet Request
Has
Message
Contact Description
Created Date
History Item
Has
Subletee Info
First Name
Last Name
Assumptions:
- Buildings in different residences can have the same name
Note: Used ___ instead of --- for partial keys

Database Schema & Functional Dependencies & BCNF Normalisation

User(Email, Password)
Primary Key: Email
Email -> Password

SubleteeInfo(FirstName, LastName, ContactDescription, Email)
Primary Key: Email
Foreign Key: (Email) refers to User
Email -> FirstName, LastName, ContactDescription

SubletRequest(PostId, SubleteeEmail, Status, Message)
Primary Key: PostId, SubleteeEmail
Foreign Key: (PostId) refers to SubletPost, (SubleteeEmail) refer to SubleteeInfo
PostId, SubleteeEmail -> Status, Message

HistoryItem(PostId, SubleteeEmail, CreatedDate)
Primary Key: PostId
Foreign Key: (PostId) refers to SubletPost, (SubleteeEmail) refer to SubleteeInfo
PostId -> SubleteeEmail, CreatedDate

SubletPost(PostId, Price, StartDate, EndDate, AdditionalInfo, Status, Building, ResidenceName, RoomNumber, SubletterEmail)
Primary Key: PostId
Foreign Key: (Building, ResidenceName, RoomNumber) refers to Room, (SubletterEmail) refers to User
Candidate Key: (SubletterEmail, StartDate, EndDate)
PostId -> Price, StartDate, EndDate, AdditionalInfo, Status, Building, ResidenceName, RoomNumber, SubletterEmail
SubletterEmail, StartDate, EndDate -> PostId, Price, AdditionalInfo, Status, Building, ResidenceName, RoomNumber
Here, SubletterEmail, SDate, EDate together is a Candiate Key and PostId is the Primary Key, so the Table is in already in BCNF and 3NF.

Room(RoomNumber, Building, ResidenceName, Floor, GenderRestriction, UnitTypeName)
Primary Key: RoomNumber, Building, ResidenceName
Foreign Key: (ResidenceName) refers to Residence, (UnitTypeName) refer to UnitType
RoomNumber, Building, ResidenceName -> Floor, GenderRestriction, UnitTypeName
RoomNumber -> Floor

RoomNumber is not a CK (or SK)
Decompose this FD to R1(RoomNumber, Floor) and R2(RoomNumber, Building, ResidenceName, GenderRestriction, UnitTypeName)

In R1, Number is a PK, so now R1 is a BCNF.
So, we will have R1 and R2 as Normalized Tables as R1 is now in 3NF and BCNF because now Number is a Candidate Key for R1.

Residence(ResidenceName, PictureLink, MoreInfoLink)
Primary Key: ResidenceName
Candidate Key: MoreInfoLink, PictureLink
ResidenceName -> PictureLink, MoreInfoLink
PictureLink -> MoreInfoLink, ResidenceName
MoreInfoLink -> ResidenceName, PictureLink
(We assume there are a distinct picture and info link for each Residence so when we get a link we know which residence it refers to)
They are both in 3NF and BCNF as the LHS in both are CKS. Name on the RHS of both is also the Primary Key of Residence.

UnitType(UnitTypeName, HasKitchen, NumberOfResidents, NumberOfBathrooms, FloorPlanLink)
Primary Key: UnitTypeName
UnitTypeName -> HasKitchen, NumberOfResidents, NumberOfBathrooms, FloorPlanLink

OriginalPrice(ResidenceName, UnitTypeName, YearRoundPrice, WinterSessionPrice)
Primary Key: ResidenceName, UnitTypeName
Foreign Key: (ResidenceName), refers to Residence, (UnitTypeName) refer to UnitType
ResidenceName, UnitTypeName -> YearRoundPrice, WinterSessionPrice

SQL DDL

```
CREATE TABLE User
(
      email VARCHAR(50),
      password VARCHAR(10) NOT NULL,
      PRIMARY KEY(email)
)

CREATE TABLE Subletee_info
(
      email VARCHAR(50),
      first_name VARCHAR(20) NOT NULL,
      last_name VARCHAR(20) NOT NULL,
      description TEXT,
      PRIMARY KEY(email),
      FOREIGN KEY(email) REFERENCES Subletee
          ON DELETE CASCADE
)

CREATE TABLE Request
(
      email VARCHAR(50),
      post_id VARCHAR(10),
      status VARCHAR(7) NOT NULL,
      message TEXT,
      PRIMARY KEY(email, post_id),
      FOREIGN KEY(email) REFERENCES Subletee_info
          ON DELETE NO ACTION
          ON UPDATE CASCADE,
      FOREIGN KEY(post_id) REFERENCES Post
        ON DELETE CASCADE
)

CREATE TABLE History
(
      email VARCHAR(50) NOT NULL,
      post_id VARCHAR(10),
      date_created VARCHAR(10) NOT NULL,
      PRIMARY KEY(post_id),
      FOREIGN KEY(post_id) REFERENCES Post
          ON DELETE CASCADE,
      FOREIGN KEY(email) REFERENCES Subletee_info
          ON DELETE NO ACTION
          ON UPDATE CASCADE
)
```

```
CREATE TABLE Post
(
      post_id VARCHAR(10),
      price DECIMAL(5,2) NOT NULL,
      start_date VARCHAR(10) NOT NULL,
      end_date VARCHAR(10) NOT NULL,
      additional_info TEXT,
      status VARCHAR(7) NOT NULL,
      building VARCHAR(20) NOT NULL,
      residence VARCHAR(20) NOT NULL,
      room_number INT(5) NOT NULL,
      email VARCHAR(50) NOT NULL,
      PRIMARY KEY(post_id),
      FOREIGN KEY(building, residence, room_number) REFERENCES Room
          ON DELETE CASCADE,
      FOREIGN KEY(email) REFERENCES Subletee_info
          ON DELETE NO ACTION
          ON UPDATE CASCADE,
      UNIQUE(email, start_date, end_date)
)

CREATE TABLE Room
(
      room_number INT(5),
      building VARCHAR(20),
      residence VARCHAR(20),
      floor INT(2),
      gender_restriction VARCHAR(20),
      unit_type VARCHAR(20),
      PRIMARY KEY(room_number, building, residence),
      FOREIGN KEY(residence) REFERENCES Residence
          ON DELETE CASCADE,
      FOREIGN KEY(unit_type) REFERENCES Unit_type
          ON DELETE NO ACTION
          ON UPDATE CASCADE
)

CREATE TABLE Residence
(
      residence_name VARCHAR(20)
      picture_link VARCHAR(100),
      more_info_link VARCHAR(100),
      PRIMARY KEY(residence_name)
)

CREATE TABLE Unit_type
```

```
(
	unit_type_name VARCHAR(20),
	has_kitchen VARCHAR(3),
	residents INT(1),
	bathrooms INT(1),
	floor_picture_link VARCHAR(100),
	PRIMARY KEY(unit_type_name)
)

CREATE TABLE Original_price
(
	residence VARCHAR(20),
	unit_type VARCHAR(20),
	year_round_price DECIMAL(4,2)
	winter_session_price DECIMAL(4,2)
	PRIMARY KEY(year_round_price, winter_session_price)
)
```