

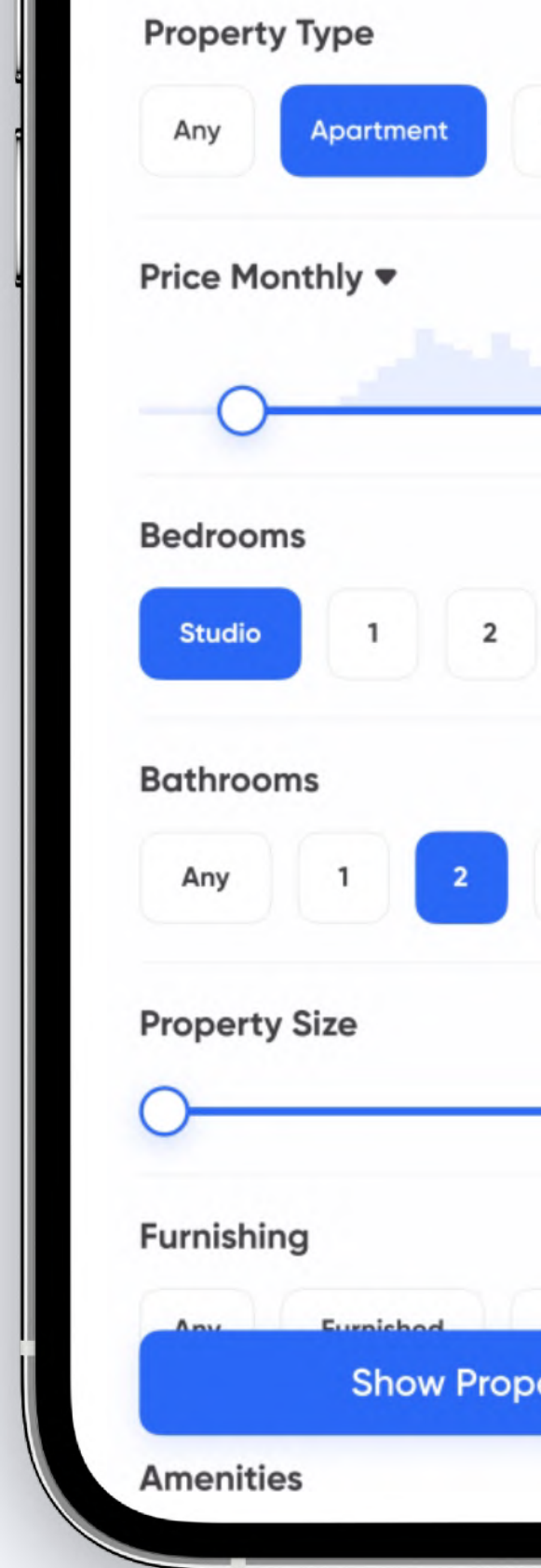
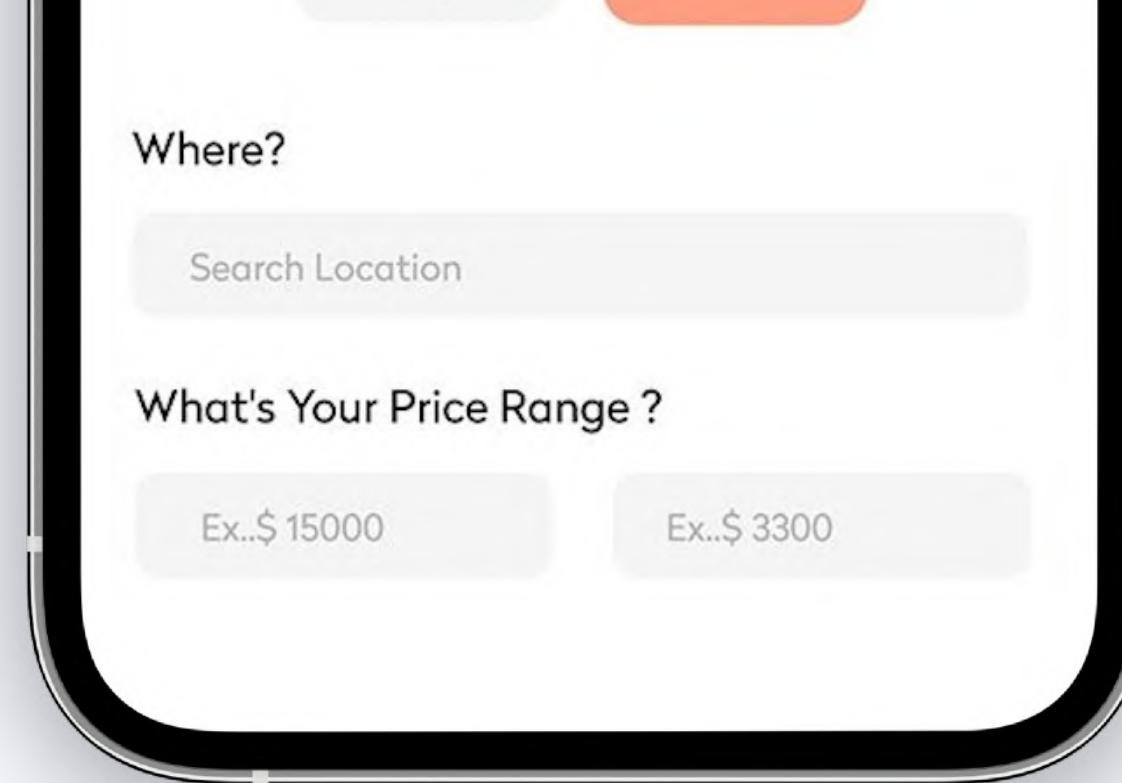
CSC 4480
Database Final Project

Project Team Members
Ifeanyi Ibeanusi
Holly Ouyang
Minh Phung
Terry Yuan

Date
April 25, 2022

Space Surfing



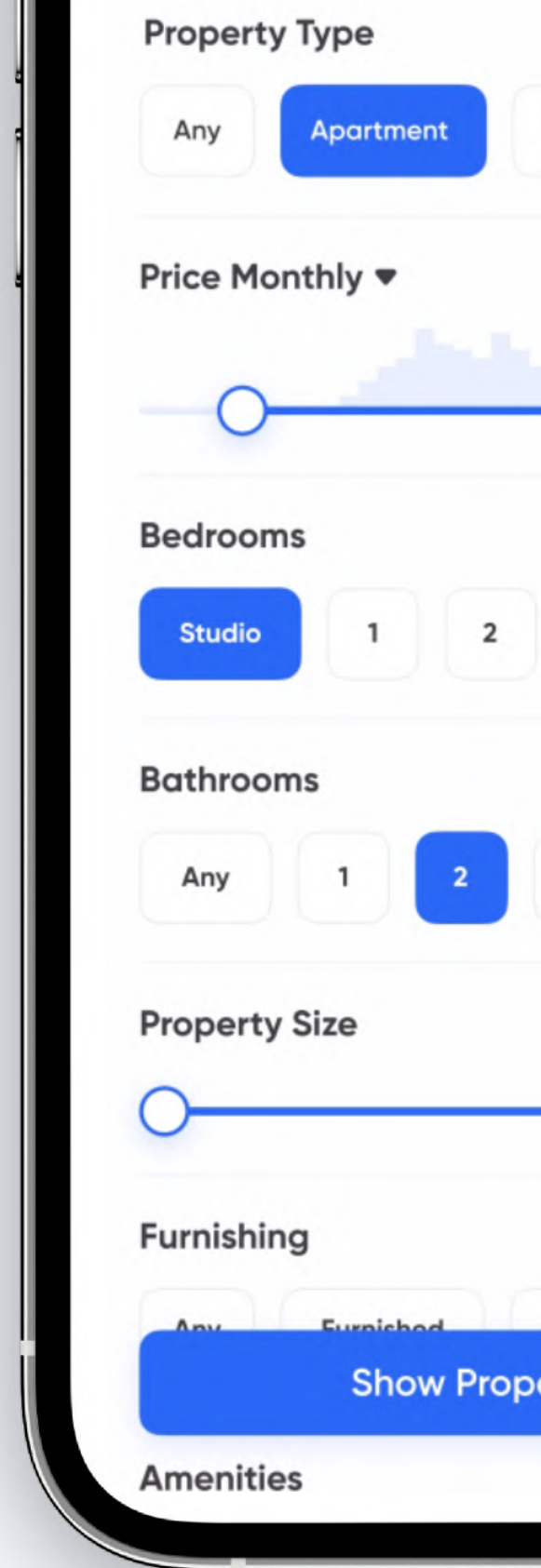
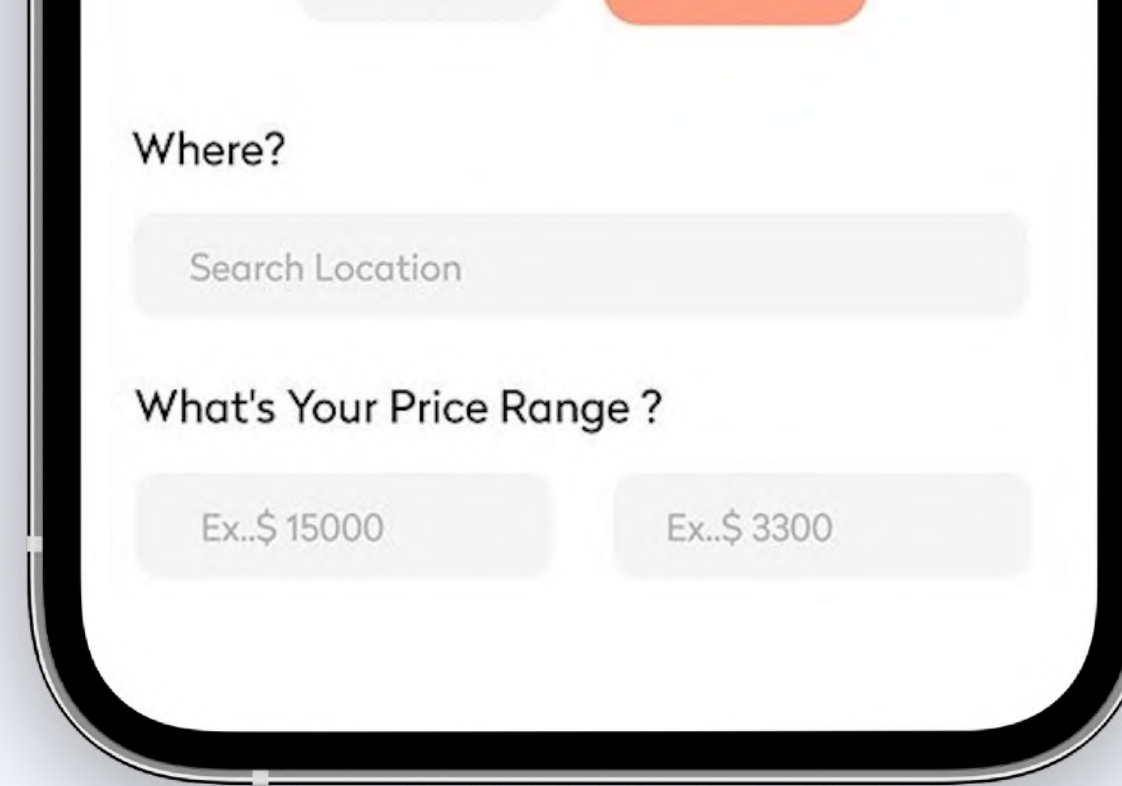


Project Summary:

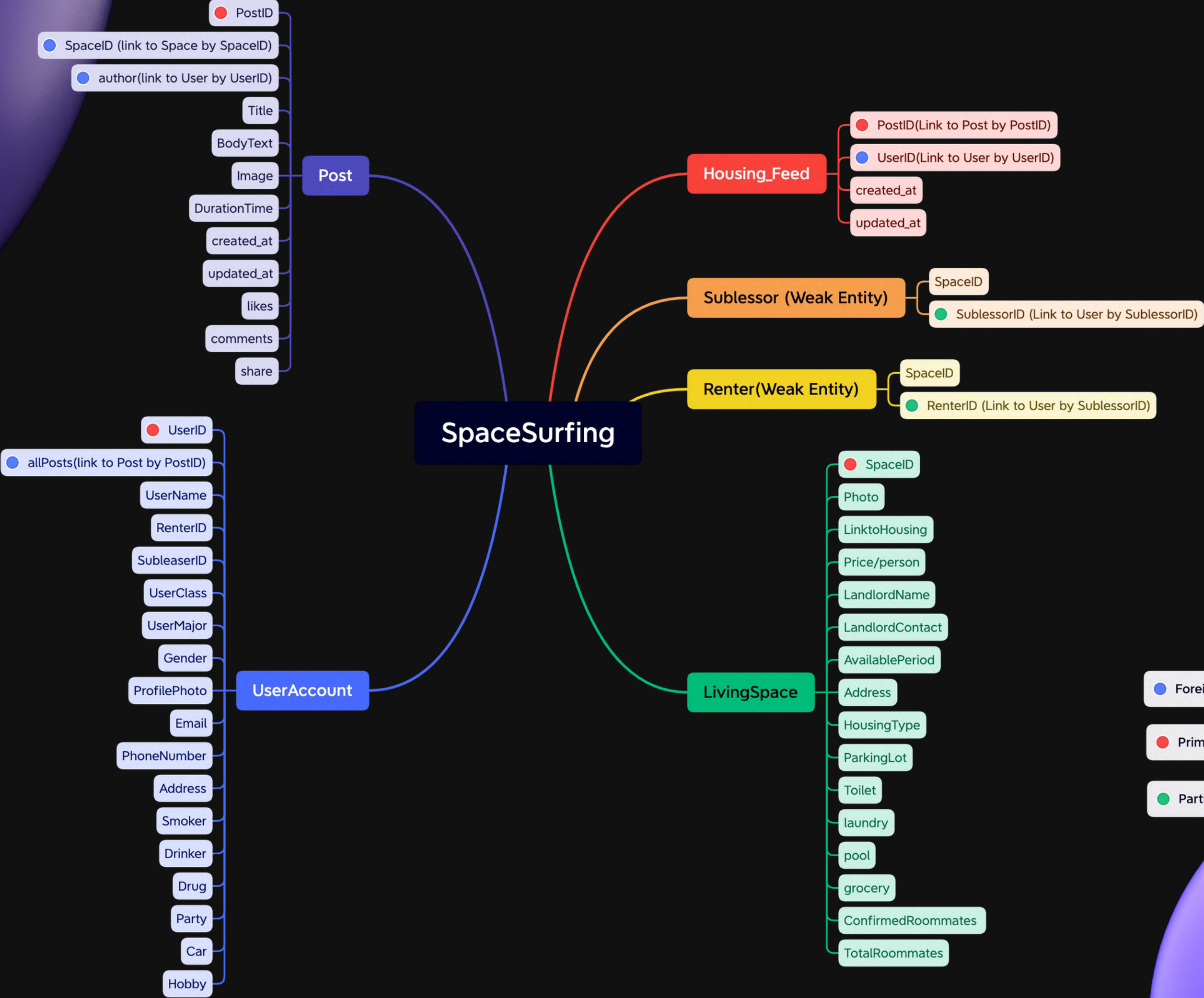
SpaceSurfing is an application that helps Villanova students look for off-campus housing while finding the most compatible roommates to live with. It allows students to be a **renter**, a **sublessor**, or both of a “space” based on their needs.

Project Summary:

“**Sublessors**” can post information about a listed property from other housing websites on this application to find preferred “renters”. Backed up by a database system, “renters” can use a filter mechanism on the housing feed to set their living preferences, browse details of available housing options, and see potential roommates' information.



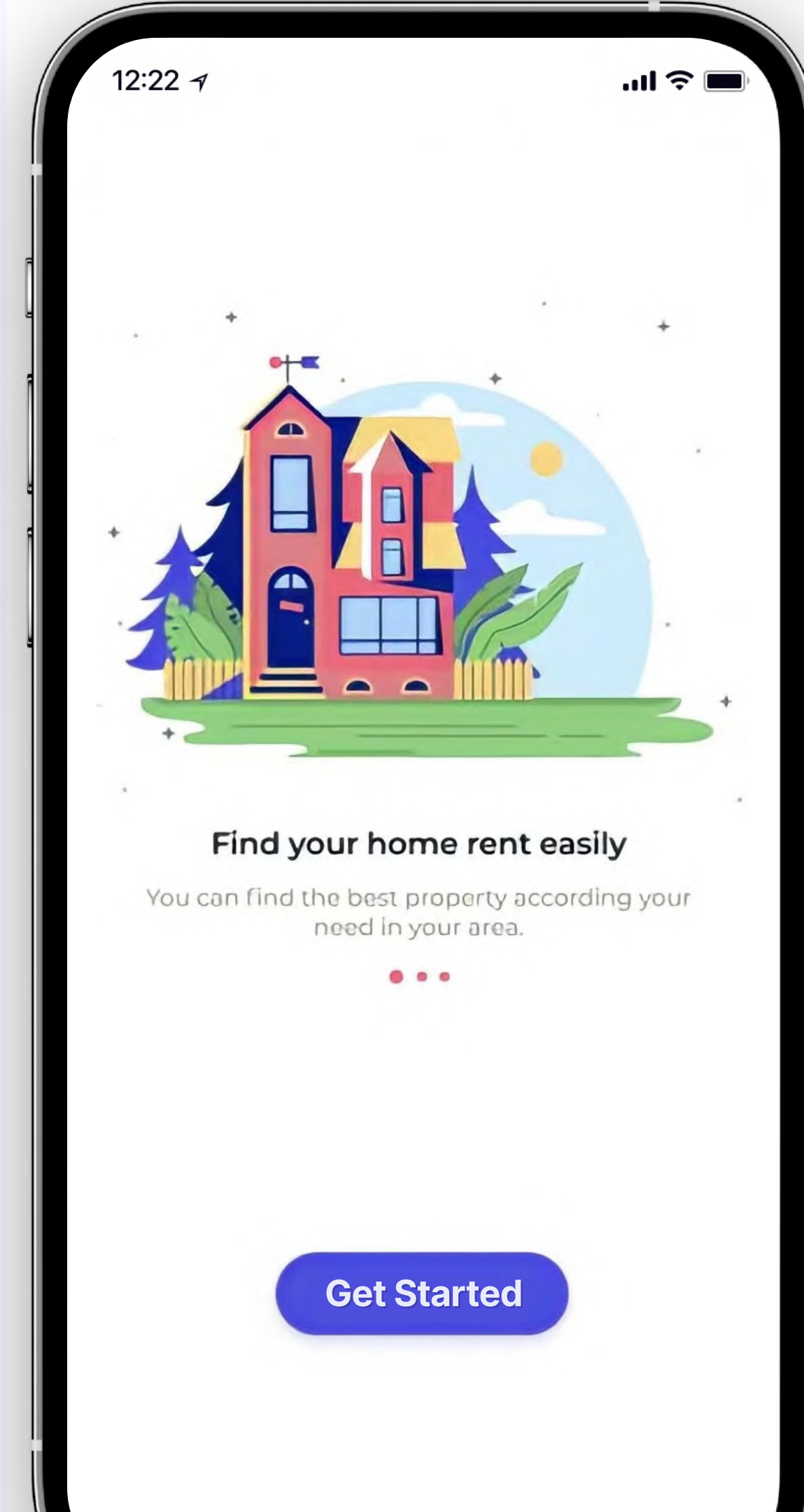
SpaceSurfing Mindmap



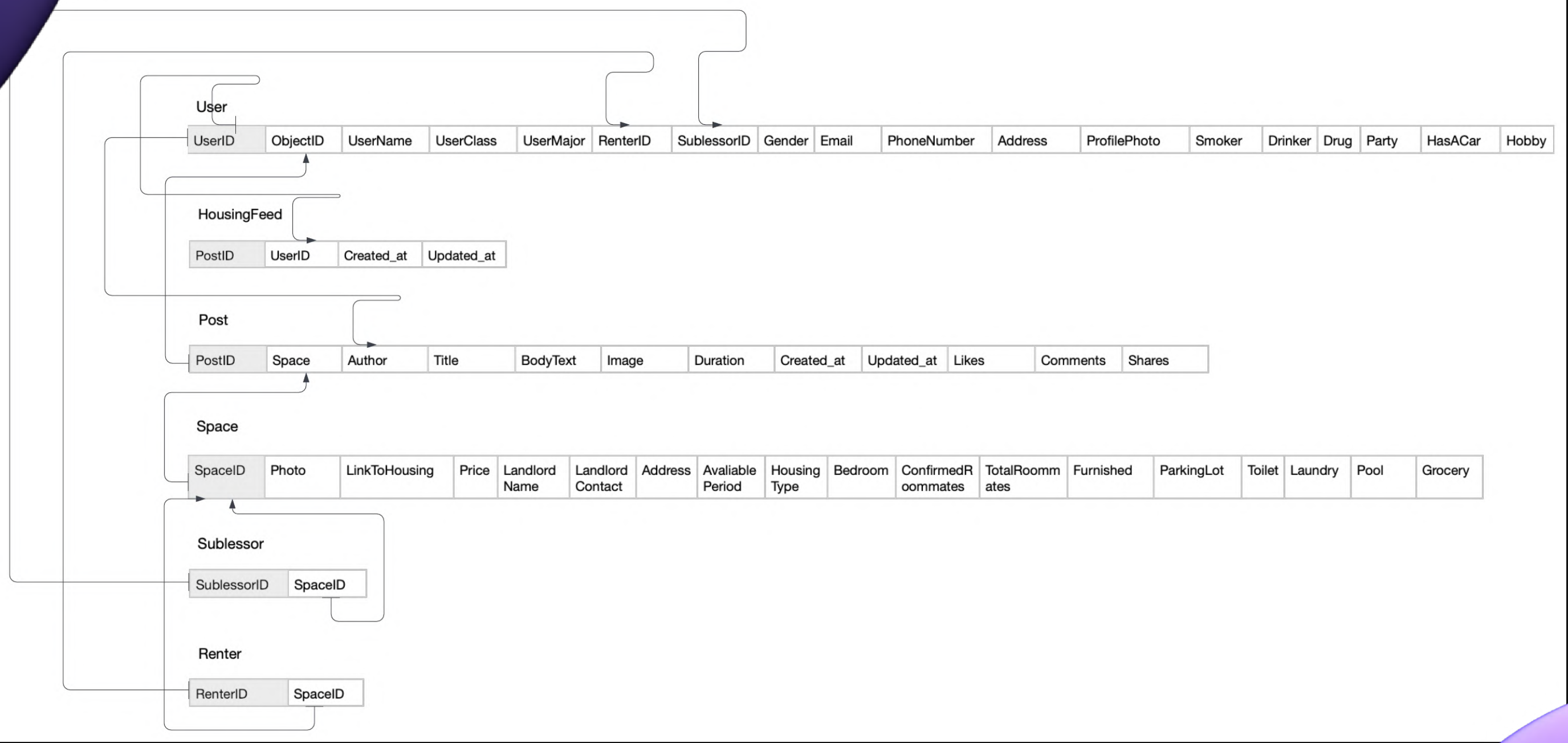


Business Rules and Restrictions

- A **HousingFeed** can have many **Posts**
- A **Post** can have 0 or 1 **Space**, a **Space** can appear in many **Posts**
- A **Post** can have 1 **Sublessor**, a **Sublessor** can do many **Posts**
- A **Sublessor** can sublease many **Spaces**
- A **Renter** can rent one **Space**
- A **User** can have 0 or 1 **sublessorID**
- A **User** can have 0 or 1 **renterID**

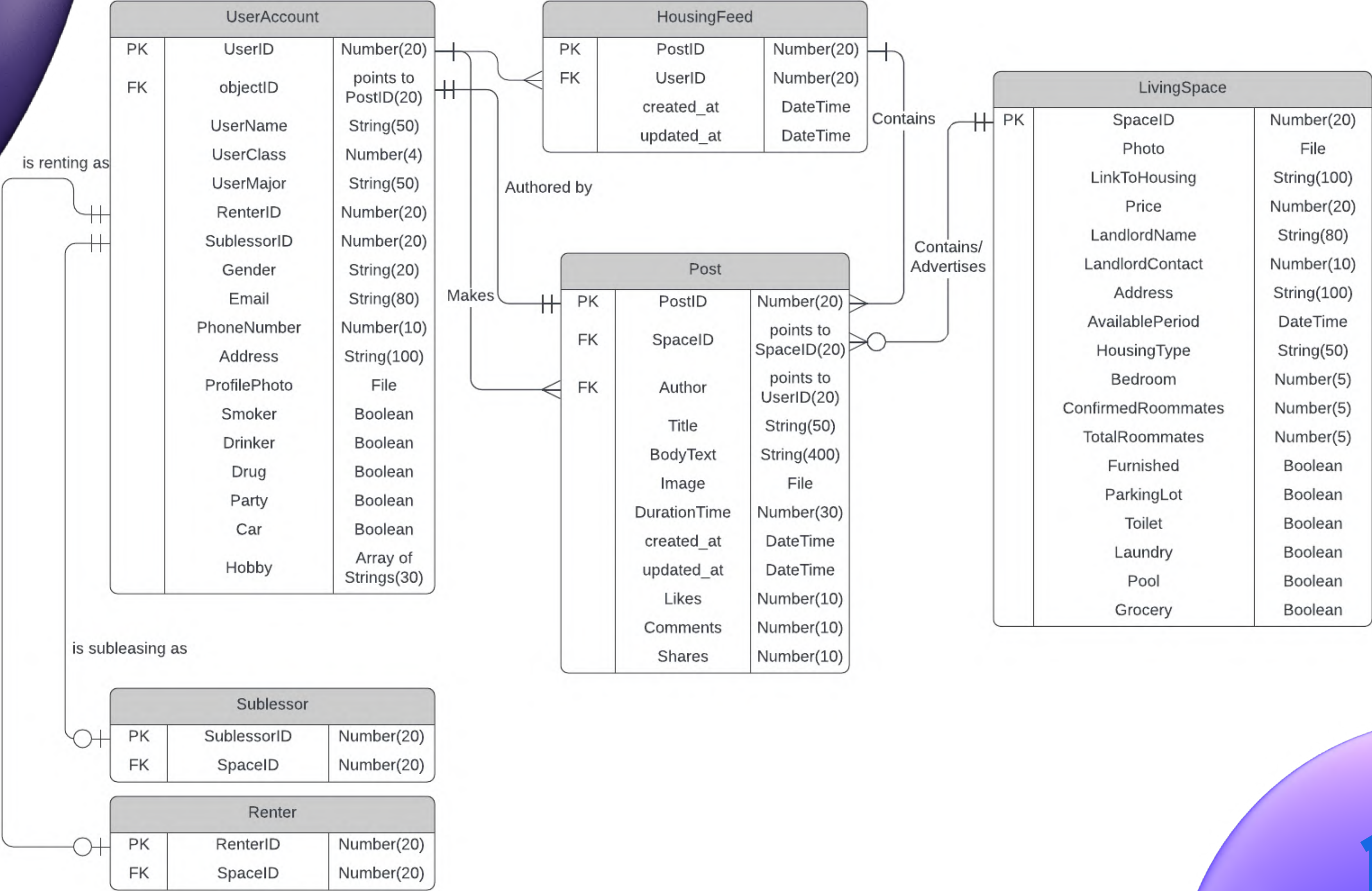


SpaceSurfing Schema



SpaceSurfing

ER Diagram



SpaceSurfing Database

UserAccount Table

SMOKER	DRINKER	DRUG	PARTY	CAR	HOBBY
0	1	1	0	1	music
0	1	0	0	0	sports
0	0	1	0	1	sing
0	0	0	1	1	dance
1	0	0	0	0	gaming
1	0	1	0	1	study
1	1	0	0	1	research
1	0	0	1	1	eating
0	0	0	0	0	skiing
0	0	0	0	0	badminton
1	1	1	1	0	fencing

```
CREATE TABLE userAccount (  
  userid      number(20) not null,  
  objid       number(20),  
  username    varchar2(50) not null,  
  userclass   number(4),  
  usermajor   varchar2(50),  
  renterid    number(20),  
  sublessorid number(20),  
  gender      char,  
  email       varchar(50),  
  phonenumber number(10),  
  address     varchar2(100),  
  profilephoto blob,  
  smoker      number(1),  
  drinker     number(1),  
  drug        number(1),  
  party       number(1),  
  car         number(1),  
  hobby       varchar2(100),  
  primary key (userid)  
);
```

	USERID	OBJID	USERNAME	USERCLASS	USERMAJOR	RENTERID	SUBLESSORID	GENDER	EMAIL	PHONENUMBER	ADDRESS	PROFILEPHOTO
1	0	(null)	James	2024	Computer En...	7	7	m	james@gmail.com	1234567893	450 Stone, Houston	(null)
2	1	(null)	Dayton	2025	Civil Engin...	6	6	m	dayton@gmail.com	1234567980	638 Voss, Houston	(null)
3	2	(null)	Ashley	2023	Chemical En...	5	5	f	ashley@gmail.com	1234569878	291 Berry, Bellaire	(null)
4	3	(null)	Diana	2023	Business	4	4	f	diana@gmail.com	9876543215	123 Peachtree, Atlanta	(null)
5	4	(null)	Franklin	2022	Business	3	3	m	franklin@gmail.com	3334455551	4333 Pillsbury, Milwaukee	(null)
6	5	(null)	Jennifer	2023	Psychology	2	2	f	jennifer@gmail.com	9876543325	7676 Bloomington, Sacramento	(null)
7	6	(null)	Jared	2025	Computer Sc...	1	1	m	jared@gmail.com	9876545641	890 Baring Street, Philadelphia	(null)
8	7	(null)	Nick	2023	Business	0	0	m	nick@gmail.com	5827902251	330 Baring Street, Philadelphia	(null)
9	8	(null)	Adam	2023	Mechanical ...	8	8	m	adam@gmail.com	5827902261	3321 Castle, Spring	(null)
10	9	(null)	Tom	2024	Biology	9	9	m	tom@gmail.com	5827902271	250 Spring Mill Road, Philadelphia	(null)
11	10	(null)	Ahmad	2025	Chemistry	10	10	m	ahmad@gmail.com	7538476221	980 Dallas, Houston	(null)



SpaceSurfing Database

HousingFeed Table

```
CREATE TABLE housingfeed (  
  postid      number(20) not null,  
  userid      number(20) not null,  
  created_at   date not null,  
  updated_at   date,  
  primary key (postid)  
  
);
```

	⚡ POSTID	⚡ USERID	⚡ CREATED_AT	⚡ UPDATED_AT
1	0	2	01-JAN-22	05-MAY-22
2	1	3	01-JAN-22	(null)
3	2	6	02-JAN-22	03-JAN-22
4	3	8	02-FEB-22	03-FEB-22
5	4	7	02-MAR-22	03-MAR-22
6	5	10	17-MAY-22	18-MAY-22

SpaceSurfing Database

Post Table

```
CREATE TABLE post (  
  postid      number(20) not null,  
  spaceid     number(20) not null,  
  author      number(20) not null,  
  title       varchar2(50),  
  bodytext    varchar2(400),  
  image       blob,  
  durationTime number(30),  
  created_at  date not null,  
  updated_at  date,  
  likes       number(10) not null,  
  comments    number(10) not null,  
  shares      number(10) not null,  
  primary key (postid)  
);
```

	POS...	SPACEID	AUTHOR	TITLE
1	0	5	0	Sublet 450 Stone Street, Sacramento
2	1	0	1	Sublet 638 Voss Street, Sacramento
3	2	1	2	Sublet 291 Berry Street, Atlanta
4	3	2	3	Sublet 123 Peachtree Street, Houston
5	4	3	5	Sublet 789 Baring Street, Philadelphia
6	5	4	9	Sublet 250 Spring Mill Road, Philadelphia

BODYTEXT	IMAGE	DURATIONTIME	CREATED_AT	UPDATED_AT	LIKES	COMMENTS	SHARES
this is text	(null)	(null)	02-APR-22	04-APR-22	15	3	6
this is text1	(null)	(null)	03-APR-22	05-APR-22	16	4	5
this is text2	(null)	(null)	07-APR-22	09-APR-22	10	2	3
this is text3	(null)	(null)	09-APR-22	13-APR-22	11	15	12
this is text4	(null)	(null)	09-MAY-22	12-MAY-22	9	2	1
this is text5	(null)	(null)	21-MAR-22	27-MAR-22	9	6	0

SpaceSurfing Database

LivingSpace Table

```
CREATE TABLE livingSpace (  
  spaceid          number(20) not null,  
  photo            blob,  
  linkToHousing    varchar2(100),  
  price            number(20,2) not null,  
  landlordName     varchar2(80),  
  landlordContactNum number(10),  
  address          varchar2(100) not null,  
  availablePeriod  date,  
  housingType      varchar2(50) not null,  
  bedroom          number(5),  
  confirmedRoommates number(5),  
  totalRoommates   number(5),  
  furnished        number(1),  
  parkinglot       number(1),  
  toilet           number(1),  
  laundry          number(1),  
  pool             number(1),  
  grocery          number(1),  
  primary key (spaceid)  
);
```

TOTALROO...	FURNISHED	PARKINGLOT	TOILET	LAUNDRY	POOL	GROCERY
3	1	0	0	1	1	1
0	1	0	1	0	1	0
0	0	1	1	0	0	1
4	1	0	0	0	1	1
4	0	0	1	0	1	0
2	0	1	0	1	1	1
3	0	1	1	0	0	0
3	1	1	1	1	1	1
1	0	0	0	0	0	0
2	0	1	1	0	0	1
5	0	0	1	1	1	1

	SPACEID	PHOTO	LINKTOHOUSING	PRICE	LANDLORDNAME	LANDLOR...	ADDRESS	AVAILABLEPERI...	HOUSINGTYPE	BEDROOM	CONFIRMED...
1	0 (null)	xyz.com		800	Andy	123465789	638 Voss Street, Sacramento	31-JUL-22	apartment	4	2
2	1 (null)	xyzw.com		1000	Margaret	123465798	291 Berry Street, Atlanta	15-JUL-22	studio	2	0
3	2 (null)	abcd.com		1200	Jason	123465978	123 Peachtree Street, Houston	15-AUG-22	townhouse	3	0
4	3 (null)	ahsafo.com		1400	Scammer	111111111	789 Baring Street, Philade...	01-APR-22	mansion	5	3
5	4 (null)	uuoo.com		700	Mark	222333444	250 Spring Mill Road, Phil...	04-MAR-22	studio	3	2
6	5 (null)	poiui.com		900	Aidan	2345678910	450 Stone Street, Sacramento	02-SEP-22	apartment	2	2
7	6 (null)	null.com		1000	Athur	4564564560	7676 Bloomington, Sacramento	03-SEP-22	apartment	3	3
8	7 (null)	tuqo.com		1100	Doyle	5675675670	3321 Castle, Spring	04-SEP-22	apartment	4	2
9	8 (null)	qffdc.com		600	Jessie	4567887654	980 Dallas, Houston	07-DEC-22	apartment	2	1
10	9 (null)	fqqff.com		700	Anna	7890609876	4333 Pillsbury, Milwaukee	21-JUN-22	townhouse	5	2
11	10 (null)	mnbvc.com		1300	Max	1234987654	890 Baring Street, Philade...	19-MAY-22	mansion	6	4



SpaceSurfing Database

Renter & Sublessor Table

```
CREATE TABLE sublessor (  
  sublessorid    number(20) not null,  
  spaceid        number(20) not null,  
  primary key(sublessorid),  
  foreign key (spaceid) references livingSpace(spaceid)  
);  
  
CREATE TABLE renter (  
  renterid       number(20) not null,  
  spaceid        number(20) not null,  
  primary key(renterid),  
  foreign key (spaceid) references livingSpace(spaceid)  
);
```

	RENTERID	SPACEID
1	0	4
2	1	3
3	2	10
4	3	2
5	4	1
6	5	7
7	6	0
8	7	8
9	8	9
10	9	5
11	10	10

	SUBLESSORID	SPACEID
1	0	5
2	1	4
3	2	3
4	3	2
5	4	1
6	5	0
7	6	10
8	7	9
9	8	8
10	9	7
11	10	6

SpaceSurfing

Foreign Key

```
ALTER TABLE userAccount ADD(  
foreign key (objid)  
references post(postid)  
);  
  
ALTER TABLE userAccount ADD (  
foreign key (sublessorid)  
references sublessor(sublessorid)  
);  
  
ALTER TABLE userAccount ADD (  
foreign key (renterid)  
references renter(renterid)  
);  
  
ALTER TABLE housingfeed ADD (  
foreign key (userid)  
references userAccount(userid)  
);  
  
ALTER TABLE post ADD (  
foreign key (author)  
references userAccount(userid)  
);  
  
ALTER TABLE post ADD (  
foreign key (spaceid)  
references livingSpace(spaceid)  
);
```



Queries

Example 1

Find information about users who are from the class of 2023.

Code:

```
SELECT userid, username, usermajor, email, phonenumber, address
FROM userAccount
WHERE userclass = '2023'
ORDER BY userid;
```

Result:

	USERID	USERNAME	USERMAJOR	EMAIL	PHONENUMBER	ADDRESS
1	2	Ashley	Chemical Engineering	ashley@gmail.com	1234569878	291 Berry, Bellaire
2	3	Diana	Business	diana@gmail.com	9876543215	123 Peachtree, Atlanta
3	5	Jennifer	Psychology	jennifer@gmail.com	9876543325	7676 Bloomington, Sacramento
4	7	Nick	Business	nick@gmail.com	5827902251	330 Baring Street, Philadelphia
5	8	Adam	Mechanical Engineering	adam@gmail.com	5827902261	3321 Castle, Spring

Queries

Code:

```
SELECT housingtype, ROUND(AVG(price)) as average_price
FROM livingspace
WHERE furnished = 1
GROUP BY housingtype
ORDER BY average_price;
```

Result:

	⚡ HOUSINGTYPE	⚡ AVERAGE_PRICE
1	apartment	950
2	studio	1000
3	mansion	1400

Example 2

Find the average price (rounded values) of furnished living spaces.

Queries

Code:

```
SELECT usermajor, COUNT(*) as count_User
FROM userAccount
GROUP BY usermajor
ORDER BY count_USER DESC;
```

Result:

	USERMAJOR	COUNT_USER
1	Business	3
2	Civil Engineering	1
3	Mechanical Engineering	1
4	Computer Science	1
5	Computer Engineering	1
6	Biology	1
7	Psychology	1
8	Chemical Engineering	1
9	Chemistry	1

Example 3

Using queries to count the number of users from each major.

Queries

Code:

```
SELECT address FROM userAccount x
WHERE x.address LIKE '%Philadelphia'
UNION
SELECT address FROM livingspace y
WHERE y.address LIKE '%Philadelphia';
```

Result:

	ADDRESS
1	250 Spring Mill Road, Philadelphia
2	330 Baring Street, Philadelphia
3	789 Baring Street, Philadelphia
4	890 Baring Street, Philadelphia

Example 4

List all the addresses of Users or Living Spaces that are based in Philly

Queries

Example 5

Find the information of users who have their post uploaded to the housing feed before April 29th, 2022 .

Code:

```
SELECT *
FROM userAccount
WHERE userID in (SELECT userID
                  FROM housingfeed
                  where updated_at < '29-APR-2022');
```

Result:

```
SELECT *
FROM userAccount a, housingfeed b
WHERE b.updated_at < '29-APR-2022' AND a.userID=b.userID;
```

	USERID	OBJID	USERNAME	USERCLASS	USERMAJOR	R...	SUBL...	GENDER	EMAIL	PHONENUMBER	ADDRESS	...	SMOKER	DRINKER	DRUG
1	6	(null)	Jared	2025	Computer ...	1	1	m	jared@g...	9876545641	890 Baring	1	1	0
2	7	(null)	Nick	2023	Business	0	0	m	nick@gm...	5827902251	330 Baring	1	0	0
3	8	(null)	Adam	2023	Mechanica...	8	8	m	adam@gm...	5827902261	3321 Castle...	..	0	0	0

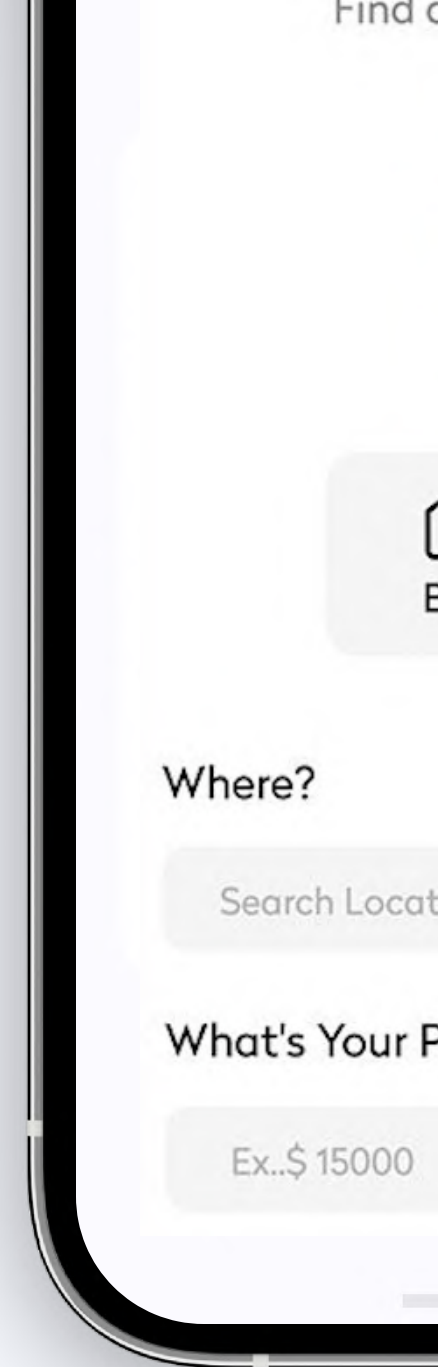
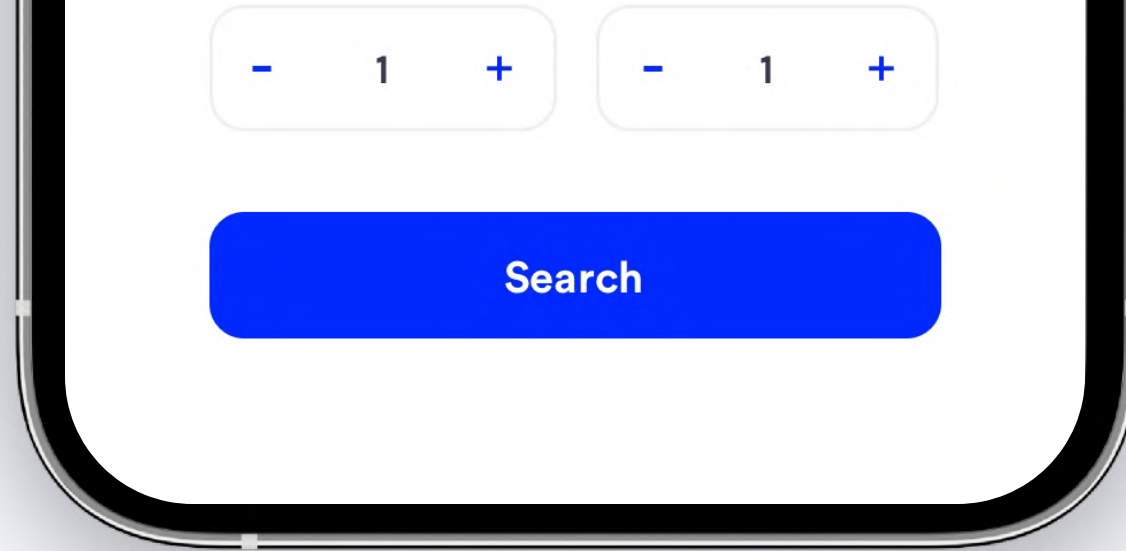
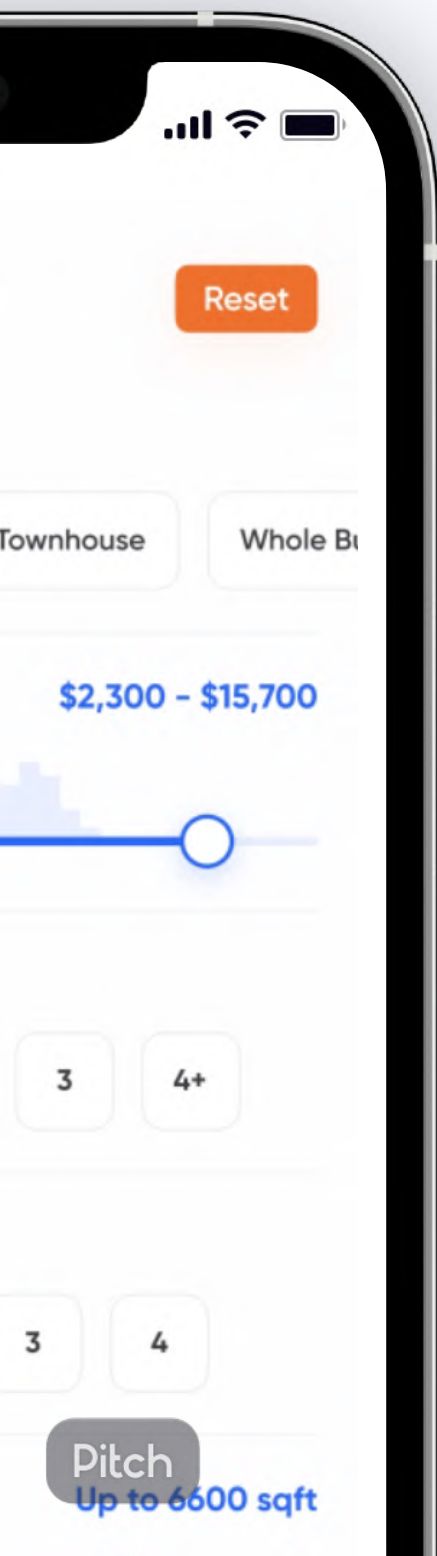
SpaceSurfing

Product Demo



Future improvements

- **Generate large volume of data** *filldb.info/dummy?*
 - Configure how to connect our app to the database server
- **Implement and upgrade User Interface & utilize APIs**
- **Research into an algorithm to determine potential matching roommate**



CSC 4480
Database Final Project

Project Team Members
Ifeanyi Ibeanusi
Holly Ouyang
Minh Phung
Terry Yuan

Date
April 25, 2022

Thank You

