Project Schema:

- **User** (user_id, password, user_name, email)
    - Primary key: user_id, an internal unique key for each row of the table
    - password: a user chosen password to ensure their identity
    - user_name: a unique way to address the user.
    - email: a way to contact the user

The User table is called user because it will hold the data for the users of our application. The user table relates to other tables because user_id will be a foreign key for many other tables. This is normalized because the data will be atomic, there isn't a composite key and there isn't any attributes functionally dependant on the non_primary key.

- **Following** (user_id, follower_id)
    - user_id and follower_id form a composite_key
    - Foreign keys: user_id and follower_id are in reference to User.user_id
    - user_id: the person doing the following
    - follower_id: the person being followed

The Following table is to determine who a user is following. The Follower table references the User table and that's it. It follows the first normal form because each attribute can have only one response. It follows the second normal form because there are only two attributes. It follows the third normal form for the same reason.

- **House** (house_id, user_id, address, price, num_bedrooms, num_bathrooms, home_type) (other possible options: hoa_fee, num_parking_spots, square_feet, lot_size, year_built, has_basement, num_stories)
    - Primary key: house_id
    - Foreign Key user_id references User
    - Column names are self explanatory

This table represents each listing that the user is interested in. Data stored in this table represent the different characteristics of each house. A user, represented by the User table, can add multiple entries to this table, and therefore recurring values for user_id will be in this table. This table follows the third normal form as data within the table is not dependent on each other. Each column of the House table is not dependent on another column.

- **Scores** (house_id, user_id, price_score, num_bedrooms_score, num_bathrooms_score, etc)
    - Primary key: user_id + house_id which makes it a Composite Key.
    - Foreign Key user_id references User
    - Foreign Key house_id references House

- Column names are self explanatory

This table represents the scores that each user gives the houses they add onto their dashboard. A user, represented by the User table, will rank each characteristic of their house on a scale from 1-10, and the data will be stored here. A composite key of both user_id and house_id is needed to distinguish each row of data from the others. This table follows the third normal form as data within the table is not dependent on each other. Each column of the House table is not dependent on another column.

Evidence of normalization:

First Normal Form:

There can't be multiple values for a cell in the table, for example house price can't be 200,000 & 250,000

Second Normal Form:

No attribute can be functionally dependent on just one part of the composite key. I.e. in a table where the composite is key user_id + house_id, don't include user_name as an attribute because it can be determined by user_id alone.

Third Normal Form:

An attribute shouldn't be determined by a non-primary key attribute. This never occurs in any of our tables.

Fourth Normal Form:

The table should not have any multi-valued dependencies. This never occurs in any of our tables because every foreign key will only relate to one value in the table that it is referencing.