Group 80: Giselle Northy & Savannah Loberger
CS340 - Final Version
3/19/2019
URL: http://flip2.engr.oregonstate.edu:5556

Project Step 7 Final Draft 3/19/2019

Breakdown of functionalities on each page:

Dogs Page

From this page the user can interact with the dog entity.

- View dogs in the database
- Filter dogs in database by:
    - Location - which shelter
    - Adoption status - coming soon, adoptable, adopted
- Add new dogs to the database
- Access Dog Details page - each dog has a page with more details
    - See vaccines of the dog
    - Edit dog information
    - Add new vaccine to dog
    - Delete a vaccine record of the dog

Owners Page

From this page the user can interact with the owner entity.

- View owners in the database
- Add new owners to the database
- Access Owner Details page - each owner has a page with more details
    - See dogs belonging to that owner
    - Add a new dog with an adoption date to an owner

Shelters Page

From this page the user can interact with the shelter entity.

- View shelters in the database
- Add new shelter to the database
- Access Shelter Details page - each shelter has a page with more details
    - Edit shelter information

Vets Page

From this page the user can interact with the vet entity.

- View vets in the database

- Add new vet to the database

Vaccines Page

From this page the user can interact with the vaccine entity.

- View vaccines in the database
- Add new vaccine to the database
- Edit name of an existing vaccine

Step 6 Feedback (Not very much feedback because we were working throughout the weekend so the site was up and down)

Kirsten Wollam

Mar 10

I am really glad i attended also, and it is too bad that the whole thing was not recorded. this link does not work, but from the link in your pdf it looks like you are on the right track and hopefully this weekend is good progress for you.

Step 5 Feedback (Not very much feedback because we were working throughout the weekend so the site was up and down)

Jessica Aten

March 10

Hi Savannah - I just wanted to let you know I'm having trouble accessing your site. Maybe you guys are in the process of making some changes, I might just be trying to load it at a bad time! I look forward to seeing your progress :)

Kirsten Wollam

March 10

I also cannot access the link above.
the link in your pdf worked: http://flip2.engr.oregonstate.edu:5556 (Links to an external site.)Links to an external site.
The pages that are working look good. Are you stuck on anything in particular?

Patrick Byrne

Mar 10, 2019

Hi Giselle,

Similar to last week, I'm really enjoying the look and feel of the site.

Dogs

The Add Dog functionality works very well. I can create a new dog and the correct attributes are displayed in the table. Are you planning on adding columns for Availability and Owner since they are both included in the form?

The update feature works great! The Dogs Details Page lets me update any of the available attributes on the page. I noticed that you can change Availability here. Do you plan on having an update feature for Owner? I suppose that owners may want to remain confidential which is why the data isn't displayed on this or the main Dogs page.

The Add Vaccination Date to Dog feature is very cool as well! I like the drop-down menu for available vaccinations. I thought this was a really nice demonstration of utilizing foreign key relationships to display data in a web page. This comes across very clearly in your DDQ (Group 80 dog_db.sql file).

Vaccines

The new Vaccines page lets me add a new Vaccine, similar to last week's page. Each of the vaccines listed has a delete button next to it but, as you mentioned, the delete functionality isn't working yet. Delete was really challenging for my group to implement so I completely understand.

Anyway, I think this is going to be an awesome site when you get it up and running. I look forward to seeing it!

Andy Tran

Mar 11, 2019

Hey Giselle,

Progress looks good on the Dog page. Can't comment on the Vaccines page - I am unable to reach it from either of the links.

I've been brainstorming ways to better integrate the update link for our group's project and I like the way you've made the dog's name into the update link. The update feature worked fine for me, same with adding vaccinations. For the vaccinations, it might make

sense to have a default placeholder value to prevent the submit button from adding whatever the first vaccine is to the list.

Great work so far!

This page is our first one that works with the database. We will be adding more to it.

Dog Adoption Database Website: **http://flip2.engr.oregonstate.edu:5556/** **(OLD)**

Project Step 5 and 6: 3/8/2019

We did not change the below documents. We worked on the Dogs page -> Update Operations.

- Some of the Update Operations are functional, some are not working yet
- Worked on Vaccines and Delete, not working yet
- Added code to the DMQ File
- Database File stayed the same

Project Step 4 Draft 2/28

We did not make any changes to our design other than working on our website and getting it to work with the DB.

Project Step 3 Final Changes 2/17

Feedback by the reviewers:

Andy Tran

Wednesday Feb 13 at 8:23pm

Project Outline

Really great project outline - thoroughly documents the progress your group has made on the project. I also liked the inclusion of the database description pdf.

Project Page

Every page had appropriate forms for inserting/updating/removing entries - very comprehensive.

DDQ SQL queries

Everything appeared to be consistent with what is described in the project outline. Looks like the weight attribute for a dog entity is able to be null, is this correct?

DMQ SQL queries

All basic manipulation queries are represented. Looking at all the queries, would it also make sense to have a delete query for dogs?

Great progress! I appreciate the level of detail present in your documentation.

Patrick Byrne

Hi Giselle,
I think the project is coming along great! It's starting to materialize into an actual app that I could envision myself using. Given that I love animals and am totally into your concept, I'm very happy with your progression.

Actions based on feedback from Draft 2:
I think it was a solid idea to remove the Country option and keep the database focused on US shelters only. I also agreed with your decision to switch from dogs having at least one owner to zero or more owners. Since a main purpose of the app is to facilitate dog adoption, it makes sense to be able to identify some dogs as having zero owners until adopted. I also thought it was a good idea to adopt the suggestion to track adoption dates and to include dog birthdate, including guessing a date if the owners don't know the actual date just to keep track of ages. I appreciated all decisions made in this section with regards to suggestions from the Draft 2 review.

DDQ:
The database definition query was excellent! I imported it into phpmyadmin and every table was created correctly with the example insert data. I also appreciated the way you used Alter Table queries to identify Foreign Keys and to also add both constraints and cascade functionality. My group put the foreign key/cascade functionality into the create queries, but it was great to see this alternative way to create tables and incorporate foreign keys.

DMQ:
The database manipulation query file also worked very well. Update and Delete both worked without causing a foreign key reference error thanks to your Alter Table queries. I did have a question regarding your choice of update and delete entities. Why only include update options for the dog availability and the dog owner? Wouldn't there also be a need to update vet/shelter information? Also, I noticed that there is not a delete option for dogs. When a dog is adopted, will it be kept in the database and just flagged with adoption status or will it be removed? I suppose I could see how it would be useful to keep the dogs information after adoption but I wasn't sure if that was intentional.

I was wondering if you were planning to use joins between tables. Are you going to cross-reference tables to look for relationships? If so, the select queries could result in redundant data without using join statements.

Website:
I really enjoyed the layout of the website pages. The basic design was well done and the forms on each page made sense. I agree with another reviewer that it would be cool to add images of the dogs!

Anyway, very well done! I look forward to the final product!

Jessica Aten:

Your project looks great. I think the changes you made based on feedback during the last step all make a lot of sense and were well thought out. The website is well organized, and it looks like you included forms for the pertinent information. One comment regarding the bridging tables. An alternative to assigning each row a primary key would to be use the two foreign keys as a composite key. I just went through my own draft and changed this for our many to many relationship tables. One tiny design note is that maybe adding a margin to your body would be nice for readability. I look forward to seeing your final project.

Kirsten Wollam

All the updates look great and your website looks clear and easy to follow.
I would take a look at your Schema, it seems like some of the arrows may be pointing the wrong directions. (specifically, dog ID to Dog_owner, i think Dog_Owner is referencing Dog ID not the other way around. Same for Owner ID and Dog_Owner).

On the updated ERD i think either the line from Owners to Dogs should be removed or the Dog_Owners table should be removed. Dogs only have either a connection to Dog_Owners or Owners, not both. Bridging tables can be left off the ERD, but if it is kept i would think the connection should only be shown through it.
For the website, you have several places where things can be modified by putting in the name of something, but your specs don't state that names must be unique. what if two dogs are named Fido? Also, your code takes an ID, not a name for the queries you wrote. maybe you can present it as a drop down with ID - Name? there as code in the sample provided to create this query.

Your DMQ code is also missing any queries that relate to your bridging tables. at a minimum insert and select functions will be needed and potentially a way to delete these also, the spec for the project is a bit vague on that.

Actions based on the Step 3 feedback 2/17:

- An alternative to assigning each row a primary key would to be use the two foreign keys as a composite key. I just went through my own draft and changed this for our many to many relationship tables.
    - We appreciated this suggestion, but aren't sure if it is necessary for our project.

- One tiny design note is that maybe adding a margin to your body would be nice for readability. I look forward to seeing your final project.
    - Fixed margins on the website body.

- I would take a look at your Schema, it seems like some of the arrows may be pointing the wrong directions. (specifically, dog ID to Dog_owner, i think Dog_Owner is referencing Dog ID not the other way around. Same for Owner ID and Dog_Owner).
    - Corrected the schema arrows on dog ID to dog_owner, and owner_id to dog_owner

- On the updated ERD I think either the line from Owners to Dogs should be removed or the Dog_Owners table should be removed. Dogs only have either a connection to Dog_Owners or Owners, not both. Bridging tables can be left off the ERD, but if it is kept i would think the connection should only be shown through it.
    - The extra line between owners and dogs has been removed.

- For the website, you have several places where things can be modified by putting in the name of something, but your specs don't state that names must be unique. What if two dogs are named Fido? Also, your code takes an ID, not a name for the queries you wrote. maybe you can present it as a drop down with ID - Name? there as code in the sample provided to create this query.
    - Made changes to the website so that the dog ID is associated with dogs on the website; allows us to account for dogs with the same name.

- Your DMQ code is also missing any queries that relate to your bridging tables. at a minimum insert and select functions will be needed and potentially a way to delete these also, the spec for the project is a bit vague on that.
    - We're working out queries between bridging tables. Possibly using JOINS.

- Looks like the weight attribute for a dog entity is able to be null, is this correct?
    - Yes, the weight is able to be NULL in case a person does not have that info. We're making weight part of modifying the dog.

- Looking at all the queries, would it also make sense to have a delete query for dogs?
    - We declined to make dog's deletable because we want to keep that a dog's data in the system in case they ever come back to a shelter.

- Why only include update options for the dog availability and the dog owner? Wouldn't there also be a need to update vet/shelter information?
    - Added the ability to update vet and shelter info.

- Also, I noticed that there is not a delete option for dogs. When a dog is adopted, will it be kept in the database and just flagged with adoption status or will it be removed? I suppose I could see how it would be useful to keep the dogs information after adoption but I wasn't sure if that was intentional.
    - We declined to make dog's deletable because we want to keep that a dog's data in the system in case they ever come back to a shelter.

- I was wondering if you were planning to use joins between tables. Are you going to cross-reference tables to look for relationships? If so, the select queries could result in redundant data without using join statements.
    - We are considering JOINS for our bridging tables, dog_owners and dog_vaccination dates

- I agree with another reviewer that it would be cool to add images of the dogs!
    - This is something we plan to add to our website later if time allows: Allow at least 1 photo of each dog.

Upgrades to the Draft version:

Website:

- Added vet and shelter modifying to the respective pages.
- Added table as an insert for dog owners

Fixes based on Feedback from Step 1:

The only feedback we received on our outline was "Great!", therefore we didn't change anything with the Project and Database Outline. Below the outlines, are the Schema and ERD Diagram.

Feedback by the peer reviewers

Patrick Byrne - Review Group 12

Fixes based on Feedback Step 1

- N/A because according to the original grader, there was no need to fix anything. Nice!

Project Outline

- Great idea! The outline was very descriptive and gave me a strong idea of exactly what you hope to accomplish. The only question that came to mind was the scope of the shelter network. For example, is it a worldwide network?

Database Outline

- I thought that the entities were explained very clearly. I like that you considered the possibility of letters in certain zip codes and used a VARCHAR appropriately to handle that situation. I did have one question with regards to the possibility that the country could be outside the US. The STATE attribute for Shelter, Owner, and Vet has no default but cannot be blank. If they are from a country other than the US that doesn't really have states, what would they put in that field? I suppose provinces and territories could be used instead. Perhaps it should have a default just in case. Of course, if the network ends up being entirely within the US, then this point is moot.

Entity Relationship Diagram (ERD)

- I thought that the ERD was done very well. The only issue I had that stuck out was the dogs to owners relationship. The outline says that dogs to owners are a many-to-many relationship, since there must be at least one owner but there can be multiple owners. The crows foot on the owner side is a zero to one when it should be the same as the one on the dog side, one to many.
- I thought it was a really smart idea to use a composite entity table to join Dogs and Vaccinations via the Vaccination Date.

Schema

- The schema was very clear and made sense. I don't really have any criticisms of that section.

Overall this was an excellent start! This looks like it's going to be an awesome project. Honestly, I had a difficult time finding any criticisms. Keep up the good work!

Andy Tran - Review Group 12

Overall Impressions:

- Great project idea. Descriptions for the entities and attributes were clear and well written - I had very few questions that couldn't be answered by reading the outline. Diagrams were clear and well organized.

Feedback - Database Outline

- Would it be possible that the birthdate of a dog is unknown when they are brought to the shelter or is it always the case that the birthdate will be known? I'm not sure about the universe/setting that this animal shelter takes place in.
- Related to this, it may also be useful to have additional attributes to track when a dog was received at a shelter or when it was adopted.
- If your shelter is also accepting international owners - it may be nice to have something like an email address as an additional point of contact.

Feedback - Relationships

- The text for the owners to dogs relationship was a little confusing - it also doesn't seem to match what the ER diagram shows. The text says "dogs must have at least one owner" but in the outline and the ER diagram show that a dog may not have an owner in the case that it is awaiting adoption.

Feedback - Schema

- Vets table is missing the phone number column.

Other

- Cute puppy images? I don't know how images are connected to databases. But your project reminded me that it would be nice to have product images for my group's project too.

Jessica Aten - Review Group 21

This is a great idea, and everything is well organized and easy to follow.

The only part I was a little confused on is the relationship between dogs and owners in the ERD. In the outline, I read that this is a many to many relationships as a dog can end up with one or two owners. The ERD indicates a dog can have 0 or 1 owner, so this was a little unclear for me.

Kirsten Wollam - Review Group 21

This is a great idea and very well done outline. In general this is really great as is, but here are a few small things i took not of as i read it. very small nit picks on a very good outline.

- The project outline states that dogs can have the status "(coming soon, adoptable, and adopted)." I highly recommend adding the oxford comma, it saves lives. https://www.theodysseyonline.com/15-reasons-you-should-use-the-oxford-comma (Links to an external site.)Links to an external site.
- The database outline describes Dog Entity saying "represents a unique adoptable animal and its attributes." Dogs can however be "coming soon" or "adopted" as well. i would suggest just removing the word adoptable for clarity.

- Making owners of dogs a many to many relationship seems over complicated as a dog can really only live in one house. There does not seem to be a lot of value in having more than one owner for the complexity it adds.
- It seems like the ERD should come before the schema in the report. Also, the ERD is not in the format from the lectures and is a bit hard to read with all the bubbles. a simple list as shown in the lectures seems easier to follow.
- On the ERD vaccination_dates seems like a bridging table. it is a way to show the many to many relationship. per the lecture, this doesn't need to be included in either the outline or the ERD. it first shows up in the schema.

Actions based on the feedback

- The STATE attribute for Shelter, Owner, and Vet has no default but cannot be blank. If they are from a country other than the US that doesn't really have states, what would they put in that field?
    - We removed Country from all the tables that kept addresses. This database is only for shelters in the US.
- The outline says that dogs to owners are a many-to-many relationship, since there must be at least one owner but there can be multiple owners. The crows foot on the owner side is a zero to one when it should be the same as the one on the dog side, one to many.
    - Changed the relationships in our database part 3 to: A dog can have zero owners until adopted.
    - Change the ERD crow's foot notation on owners to dogs. Format was 0 to 1, change to 0 to many.
    - Changed the schema to reflect this relationship as many to many
- Would it be possible that the birthdate of a dog is unknown when they are brought to the shelter or is it always the case that the birthdate will be known?
    - Decided not to make make a change because while its likely some dogs won't have a known birthdate, it's better for users to guess a date so that the table can have a way to keep track of ages more accurately, instead of manually updating it.
- Related to this, it may also be useful to have additional attributes to track when a dog was received at a shelter or when it was adopted.
    - We decided to make this change. Not only to track the adoption dates, but also because we needed to add a bridging table for the many-to-many relationship.
- If your shelter is also accepting international owners - it may be nice to have something like an email address as an additional point of contact.
    - We decided to make this change and added the email to both the Owners and Shelters.
- The text for the owners to dogs relationship was a little confusing - it also doesn't seem to match what the ER diagram shows. The text says "dogs must have at least one owner" but in the outline and the ER diagram show that a dog may not have an owner in the case that it is awaiting adoption.

    - This is similar to another's comment above, which will be fixed in the diagram and outline wording.
    - Re-did the ERD for a better format
- Vets table is missing the phone number column.
    - This was fixed and added to the ERD and the schema.
- Cute puppy images?
    - We like this idea, but we're not sure if we'll have time for adding this yet.
- The only part I was a little confused on is the relationship between dogs and owners in the ERD. In the outline, I read that this is a many to many relationships as a dog can end up with one or two owners. The ERD indicates a dog can have 0 or 1 owner, so this was a little unclear for me.
    - Similar comment from others that we fixed
- The project outline states that dogs can have the status "(coming soon, adoptable and adopted)." I highly recommend adding the oxford comma, it saves lives. https://www.theodysseyonline.com/15-reasons-you-should-use-the-oxford-comma (Links to an external site.)Links to an external site.
    - Fixed this in the outline
- The database outline describes Dog Entity saying "represents a unique adoptable animal and its attributes." Dogs can however be "coming soon" or "adopted" as well. i would suggest just removing the word adoptable for clarity.
    - We decided to make this change since it just adds to the clarity.
- Making owners of dogs a many to many relationship seems over complicated as a dog can really only live in one house. There does not seem to be a lot of value in having more than one owner for the complexity it adds.
    - We decided not to make this change. If we limit the amount of owners, we can only track info for one owner on a dog. What if the dog gets a new owner later? Or a couple adopts a dog and they both want to be listed as owners? We would be able to see history of owners for a dog. Also, we added a bridging table in this new version to make it clearer and more readable.
- It seems like the ERD should come before the schema in the report. Also, the ERD is not in the format from the lectures and is a bit hard to read with all the bubbles. a simple list as shown in the lectures seems easier to follow.
    - We decided to make these changes. We fixed the ERD to be easier to read and put the ERD before the Schema in the document.
- On the ERD vaccination_dates seems like a bridging table. it is a way to show the many to many relationship. per the lecture, this doesn't need to be included in either the outline or the ERD. it first shows up in the schema.
    - We decided not to make this change. We kept the vaccination_dates bridging table because it made sense to us to leave it. There is also another piece of information so it isn't just a foreign key table, the vaccination date.

Upgrades to the Draft version

- In addition to the ideas from the comments above, we added email addresses to Shelters as well for another way to contact.

- We also added one bridging table - between the Dogs to Owners. This table also holds the adoption date of the dog.

Group 80: Giselle Northy & Savannah Loberger
CS340 - Project Step 2 Final Version: Revisions, Outline, ERD, & Schema
2/3/2019

Dog Adoption Database

Project Outline

The database we will be making will represent a dog adoption shelter network. This is for a fictional organization but could be used in real life to track adoptable dog attributes and status such as their name, age, breed, weight, location, status (coming soon, adoptable, and adopted) and new owner information. In the network, there are multiple shelters and new shelters can be added later. There will also be vets that are tracked at the shelters. It is assumed that when dogs are entered into the system, they will have some unknown attributes that can be entered later; noted below where needed. When dogs are adopted, they remain in the system and will then need owner information.

Database Outline

The entities in the database are:

Dogs – Dogs are the most important entity because it has relation to all other entities in the database. Dogs entity represents a unique animal and its attributes.

- **ID**: This number is automatically assigned to each dog when they are added to the database. This is an auto-incrementing number which is the primary key.
- **Name**: Name of the individual dog is a VARCHAR with a maximum of 100 characters. It cannot be blank and there is no default.
- **Birth Date**: Birth date is tracked by birth date and uses the SQL type DATE (YYYY-MM-DD). This cannot be blank and there is no default.
- **Gender**: Gender is of VARCHAR type and the choices are “Male” or “Female”. This cannot be blank and there is no default.
- **Breed**: Breed is the type of dog and is a VARCHAR with a maximum of 100 characters. Because there are so many different dog breeds, an attempt to classify them all won’t be made and instead a user can enter the breed. This cannot be blank, but the default is “Unknown.”
- **Color**: Color will be a VARCHAR with a maximum of 100 characters. Same as breed, there are too many variations to an attempt won’t be made to classify them. This cannot be blank, but the default is “Unknown.”
- **Weight**: Weight is a SMALLINT with a max of 3 characters that is measured in lbs. That is the max weight because no dog would be bigger than that.
- **Availability**: This is type VARCHAR and represents the dog’s status. The choices are “Coming soon”, “Adoptable”, and “Adopted”. The default is “Coming soon.”
- **Shelter ID**: This contains the unique ID of the shelter where the dog is. After being adopted, this stays so keep record of where the dog came from. When Shelter ID is NULL, it removes the relationship to the Shelter (the dog was adopted).

- **Owner ID**: This is the unique ID of the person who adopts the dog. Defaults to NULL because dogs don't have an owner until adopted.

Shelter – Shelters are important because they hold the address and contact information for an individual dog shelter. These have relationships with Dog entities.

- **ID**: This is an auto-incrementing number which is the PRIMARY KEY for Shelter types. Because it auto increments, additional shelters can be added later.
- **Name:** Name is type VARCHAR with a max of 100 characters which represents the name of the shelter. It cannot be blank and there is no default.
- **Address:** Address is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **City:** City is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **State:** State is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **Zip code:** Zip code is stored as type VARCHAR because there are some places that have letters in the zip code. It cannot be blank and there is no default.
- **Phone #:** Phone number is type VARCHAR and it cannot be blank. There is no default.
- **Email:** Email address is stored as type VARCHAR with a max of 100 characters. It can be blank and there is no default.

Owner – Owners holds information on the owner that adopts a dog. This has a relationship with dog entities.

- **ID**: This is an auto-incrementing number which is the PRIMARY KEY for Owner types. Because it auto increments, additional owners can be added later.
- **First Name:** Name is type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **Last Name:** Name is type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **Address:** Address is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **City:** City is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **State:** State is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **Zip code:** Zip code is stored as type VARCHAR because there are some places that have letters in the zip code. It cannot be blank and there is no default.
- **Phone #:** Phone number is type VARCHAR and it cannot be blank. There is no default.
- **Email:** Email address is stored as type VARCHAR with a max of 100 characters. It can be blank and there is no default.

Vet – Vets hold information of the veterinarian that works at a shelter. This has a relationship with shelter entities.

- **ID**: This is an auto-incrementing number which is the PRIMARY KEY for Vet types. Because it auto increments, additional vets can be added later.

- **First Name:** Name is type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **Last Name:** Name is type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **Address:** Address is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **City:** City is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **State:** State is stored as type VARCHAR with a max of 100 characters. It cannot be blank and there is no default.
- **Zip code:** Zip code is stored as type VARCHAR because there are some places that have letters in the zip code. It cannot be blank and there is no default.
- **Phone #:** Phone number is type VARCHAR and it cannot be blank. There is no default.
- **Shelter ID**: This contains the unique ID of the shelter where the vet works.

Vaccination Types – Vaccinations hold the names of the types of vaccines.

- **ID**: This is an auto-incrementing number which is the PRIMARY KEY for Vaccination types. Because it auto increments, additional vaccinations can be added later.
- **Vaccine Name**: This is the attribute that holds the name of a vaccination. It is type VARCHAR with a max of 100 characters.

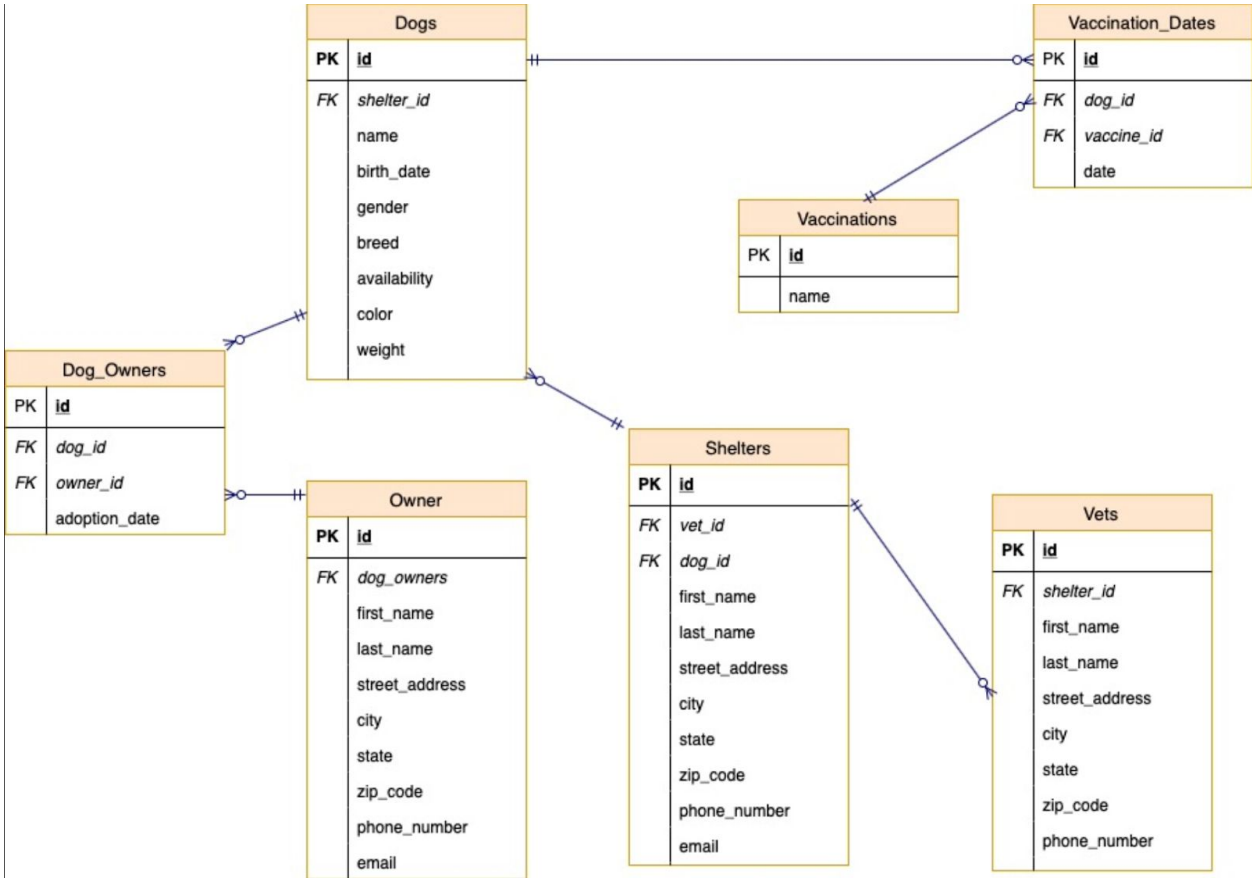
Dates of Vaccinations - This is the dates that the dogs received vaccines.

- **ID**: This is an auto-incrementing number which is the PRIMARY KEY for Vaccination types. Because it auto increments, additional vaccinations can be added later.
- **Date**: Date is when the vaccination was given and uses the SQL type DATE (YYYY-MM-DD). This cannot be blank and there is no default.
- **Dog ID:** This contains the unique ID of the dog who received the vaccine.
- **Vaccine ID:** This contains the unique ID of the vaccine given.

The relationships in our database are:

1) **Shelters have Vets** – This is a *one-to-many relationship* because shelters can have many vets while a vet works at one shelter.
2) **Dogs live in shelters** – This is a *one-to-many relationship* because multiple dogs can be in the same shelter, but a dog can only be in one shelter.
3) **Owners dogs** – This is a *many-to-many relationship* because each dog will end up with one or two owners, but owners can adopt more than one dog. The constraints are such that, there must be at least 1 dog owner, but there can be multiple owners. Also, dogs can have zero or many owners, but owners must have at least one dog, but can have multiple dogs (no limit).
4) **Dogs have vaccinations** – This is a many-*to-many relationship* because a dog can have many vaccinations and all dogs can receive the same vaccinations.

Project ERD:


Dogs
PK id
FK shelter_id
name
birth_date
gender
breed
availability
color
weight
Vaccination_Dates
PK id
FK dog_id
FK vaccine_id
date
Vaccinations
PK id
name
Dog_Owners
PK id
FK dog_id
FK owner_id
adoption_date
Owner
PK id
FK dog_owners
first_name
last_name
street_address
city
state
zip_code
phone_number
email
Shelters
PK id
FK vet_id
FK dog_id
first_name
last_name
street_address
city
state
zip_code
phone_number
email
Vets
PK id
FK shelter_id
first_name
last_name
street_address
city
state
zip_code
phone_number

Group 80: Giselle Northy & Savannah Loberger

CS340: Schema