

# UH Pet App Database Schema

## Tables/Collections

### Users

This section describes data columns for Users of the system. Data is inserted into the table/collection when the User registers through the Registration process on the website.

```
{
  _id: ObjectId, //(ObjectId is how each row is saved in MongoDB)
  email: string,
  passHash: string, //(hashed password)
  firstName: string,
  lastName: string,
  isActive: boolean, //(true = active)
  favorites: array, //(of ObjectIds)
  adoptionRequests: array //(of adoptionRequests)
}
```

### Employees

This section describes data columns for Employees of the system. Data is inserted into the table/collection as part of the Employee's onboarding process (separate process flow).

```
{
  _id: ObjectId, //(ObjectId is how each row is saved in MongoDB)
  email: string,
  passHash: string, //(hashed password)
  firstName: string,
  lastName: string,
  isActive: boolean //(true = active)
}
```

## Pets

This section describes data columns for Pets in the system. Data is inserted into the table/collection as part of the Pet registration process by an Employee.

```
{
  _id: ObjectId, //(ObjectId is how each row is saved in MongoDB)
  name: string,
  type: string
  sex: string, //(can only be one of specified enum values. ex. M or F)
  age: string, //(can only be one of specified enum values. ex. child, adult, newborn,
etc)
  size: string, //(can only be one of specified enum values. ex. small, medium, large,
extra large, etc)
  weight: double,
  dateAdded: Date, //Date and Timestamp
  description: string,
  imageNames: array, //(of string)
  isAdopted: boolean //(true = adopted)
}
```

## Adoption Requests

This section describes data columns for User requests for Pet adoptions. Data is inserted when a User requests to adopt a Pet.

```
{
  _id: ObjectId, //(ObjectId is how each row is saved in MongoDB)
  email: string,
  petId: ObjectId,
  adopterId: ObjectId,
  date: Date, //Date and Timestamp
  status: string, //(can only be one of specified enum values)
}
```

# Views

Not required by MongoDB design but supported since v3.4.

## Relationships

This section provides a pictorial representation of relations between tables/collections.

