

Phase 1 Report

Xining Li, Yi Zhao, Xiaoyan Liu, Yaguang Chen

GaTech

April 8, 2020

Abstract

This is abstract

1 Table of Contents

- [Login](#)
- [Animal Dashboard](#)
- [Add Animal](#)
- [Animal Detail](#)
- [Vaccinations](#)
- [Adoption](#)
- [Add Adoption Application](#)
- [Adoption Application Review](#)
- [Animal Control Report](#)
- [Volunteer of the Month](#)
- [Monthly Adoption Report](#)
- [Volunteer Lookup](#)
- [Vaccine Reminder Report](#)

2 Abstract Code

2.1 Login

Abstract Code

- User enters **email**, **password** input fields.
- If data validation is successful for both username and password input fields, then:
- When *Enter* button is clicked
 - If User record is found and user entered password match the username's key associated password in the **User** table: Store login information as session variable '\$UserID', and go to the **Animal Dashboard**
 - Else: Go back to the **Login**

2.2 Animal Dashboard

Abstract Code

- Show the animal's name, species, breed, sex, alteration status, age, and adoptability status by the result of the corresponding SQL query.
- Populate species and adoptability status dropdowns, if no buttons are pushed, do nothing. If clicking on species and/or adoptability status, query corresponding animals and display them on dashboard only.
- Upon:
 - Clicking on each column would pop out 2 choices: sort in increasing order, sort in decreasing order.
 - Clicking on the animal's name will go to the **Animal Detail**'s Servlet (implemented using RestFul API GET method).
- Show the number of available spaces.
- if the user has appropriate permission (fetched by the corresponding SQL query), an *Add Animal* button will show directing to the **Add Animal** screen.
- if the user has appropriate permission (fetch by the corresponding SQL query), an *Add Adoption Application* button will show directing to the **Add Adoption Application** screen.

2.3 Add Animal

Abstract Code

- Process user's post request and convert to a data structure (for example Json or POJO)
- Validate user input, if valid continue, else return user the error message.
- Acquire permit from the semaphore
- TRY:
 - Parse the animal's species from the user's post request
 - If the animal's species from the user's post request is exist in the database and the number of availability associated with the animal's species is greater than 0:
 - * Generate the unique petId by accessing the **Animal** table and pass to the setter of the petId field of the upper-mentioned data structure.
 - * Submit the data structure to the database.
 - * Take the user to the **Animal Detail** Screen.
 - Elae:

- * Return the user an error message with the corresponding error code.
- Catch Exception:
 - Log the error message and return the error message to the user.
- Finally:
 - Release the permit to the semaphore.

2.4 Animal Detail

- Show animal's all details (attributes) by the result of the corresponding SQL query.
- Show a "Vaccinations" section which shows any Vaccination history.
- Show *Add Vaccination* button if the corresponding animal has not been adopted.
- Show *Adopt pet* button if the animal is eligible for adoption and if the session user has the permission.

Abstract Code

AnimalDetailServlet

Background: Animal Dashboard's restful API takes the user to the corresponding Animal Detail's front end, and the Animal Detail's front end call the AnimalDetailServlet using the GET method. And here is the implementation of the AnimalDetailServlet's get method.

- the AnimalDetailServlet query the database with the restful api's parameter (**PetId**) and convert to a POJO
- the AnimalDetailServlet return the Json converted from the POJO to the front end.
- the AnimalDetailServlet call the AddVaccinationServlet using the GET method and return the user the vaccination that can be implemented.
- (frontend code) the user can optionally call the AddVaccinationServlet using the POST method.

AddVaccinationServlet

Background: Animal Dashboard's restful API takes the user to the corresponding Animal Detail's front end, and the Animal Detail's front end call the AddVaccinationServlet using the GET method.

```
public class Vaccination {
    @NotNull Enum VaccinationType;
    @NotNull Date date;
    @NotNull Date nextDoseDate;
    Long vaccineTagNumber;
}

public class AddVaccinationServlet extends HttpServlet {
    Set<VaccinationType> Get(Animal animal){
        synchronized (animal.getSingleton()){
            if (!animal.eligibleForAdoption()){
                return null;
            } else {
                return animal
                    .getSpecies()
                    .getRequiredVaccinations()
                    .setSubtraction(animal
                        .implementedVaccinations
                    );
            }
        }
    }
}
```

```

    }
    void Post(Set<Vaccination> userImplementations) {
        synchronized (animal.getSingleton()){
            for (Vaccination v: userImplementations) {
                /*
                 implementVaccination talks to
                 the database and put the corresponding
                 vaccination into the animal's
                 corresponding vaccination table.
                */
                DataBaseSingleton.get(animal)
                    .implementVaccination(v);
            }
        }
    }
}

```

2.5 Vaccinations

Abstract Code

- Populate vaccinations dropdown lists
- When submit button is not clicked, nothing happened
- When submit button is clicked:
 - If vaccine is not chosen: catch Exception, log error message and return the error message to the user;
 - If vaccine is chosen, not both vaccination date or next does date are entered: catch Exception, log error message to the user;
 - Else: update Animal table,
- If submit successfully, Display **Animal Detail**

2.6 Adoption

Abstract Code

- Show search dialog where user can search both applicant's last name and co-applicant's last name
- Query and display Applicant's contact information
- If select Adopter, show pop up for adoption date and adoption fee
- If enter adoption date and adoption fee, update **Adoption Information** table, display **Animal Detail**
- If cancellation, display **Animal Detail**

2.7 Add Adoption Application

Abstract Code

- Show screen to let users enter applicant information including Application first name and last name, Address (street, city, state, zip code), phone number, email address, Date of Application, (optional) co-applicant first name and last name
- When click submit button:
 - If at least one contact information is not entered: catch Exception, log error message and return the error message to the user;
 - Else: Add **Contact Information**, display application ID generated by system
- When submit button is not clicked, nothing happen

2.8 Adoption Application Review

Abstract Code

- Find applicationID with pending approval, display applicationID and status

```
SELECT application_number, status FROM ApplicationInformation
ORDER BY application_number ASC;
```

- Update status with dropdown, approved or rejected, Save application status to Contact Information table

```
UPDATE ApplicantInformation
SET status=$status
```

2.9 Animal Control Report

Abstract Code

- SuperUser clicked on *Animal Control Report* button from **Animal Dashboard**:
- Run the **Animal Control Report** task: query for animals' information.
 - Select month and find the animals that was surrendered by Animal Control in that month;
 - Sort by Pet ID ascending;
 - Display animal's information;

```
SELECT pet_id, species, breed, sex, alteration_status, microchipId,
       surrender_date
FROM ((Animal INNER JOIN Species ON Animal.pet_id = Species.pet_id)
      INNER JOIN Breed ON Animal.pet_id = Breed.pet_id)
      INNER JOIN Surrender ON Animal.pet_id = Surrender.pet_id
WHERE MONTH(surrender_date) = $selected_month AND animalcontrol = $animalcontrol
ORDER BY pet_id ASC;
```

- Find the animals that was adopted in selected month;
- Find these adopted animal's adopted date and surrendered date;
- Display animal's information if sheltered dates are greater or equal to 60 days in selected month:

```
SELECT pet_id, species, breed, sex, alteration_status, microchipId,
       DATEDIFF(day, surrender_date, adoption_date)
FROM (((Animal INNER JOIN Species ON Animal.pet_id = Species.pet_id)
        INNER JOIN Breed ON Animal.pet_id = Breed.pet_id)
        INNER JOIN Surrender ON Animal.pet_id = Surrender.pet_id)
        INNER JOIN AdoptionInformation ON Animal.pet_id = AdoptionInformation.pet_id
ORDER BY DATEDIFF(day, surrender_date, adoption_date) DESC
WHERE (DATEDIFF(day, surrender_date, adoption_date) >= 60) AND
MONTH(surrender_date) = $month;
```

2.10 Volunteer of the Month

Abstract Code

- SuperUser clicked on *Volunteer of the Month* button from **Animal Dashboard**:
- Run the **Volunteer of the Month** task: query for information and volunteered hours about the volunteers.
 - Find all volunteers that has volunteered for selected month;

- Calculate the total volunteered hours for each of these volunteers;
- Sort the list of volunteers by descending total hours volunteered for selected month and year;
- Display first 5 volunteers' first name, last name, email address and total volunteered hours in that month

```

SELECT TOP 5 SUM(hoursWorked)(first_name, last_name, email_address,
SUM(hoursWorked)
FROM LoginUser INNER JOIN VolunteerWorkHours
ON LoginUser.username = VolunteerWorkHours.username
GROUP BY username
ORDER BY SUM(hoursWorked) DESC, last_name ASC
WHERE MONTH(dateWorked) = $selected_month AND YEAR(dateWorked) =
    $selected_year;

```

2.11 Monthly Adoption Report

Abstract Code

- **find the adoption** info for this month
- **find the surrender** info for this month
- **find the animal** info according to adoption and surrender
- group by species, breed and count total
 - order by time and alphabetical order
 - combine breed for mixed animal

2.12 Volunteer Lookup

Abstract Code

- **find people** by their first name matching the search item (jon means jonson, jonstone etc.)
- **find users** by first and last name and email
- **find phone number** from volunteer table

2.13 Vaccine Reminder Report

Abstract Code

- User expiration date to **find all vaccine** that are expiring
- **Find animals** by vaccine activity id
- **Find person** who did the vaccine