



College of Engineering

## CS CAPSTONE PROGRESS REPORT

FEBRUARY 26, 2019

# APP TO SUPPORT FIELD DIAGNOSTICS IN VETERINARY MEDICINE

PREPARED FOR

## OREGON VETERINARY DIAGNOSTICS LABORATORY

PROFESSOR CHRISTIANE LOEHR

PREPARED BY

## GROUP 72 MALSANO

KATHERINE JEFFREY

BRANDON JOLLY

BRADFORD WONG

### Abstract

Currently, there are many difficulties for veterinary pathologists trying to perform remote diagnostics. There are not any effective ways for people out in the field collecting samples to communicate with specialized experts located in laboratories. As a result, this project will involve creating an android mobile application that will be used as a bridge to connect the field personnel with the veterinary pathologists in laboratories. With this mobile application, the field personnel will be able to take pictures of the individual that is being analyzed and then send the pictures along with other data such as the patient, location, and time to a pathologist. The pathologist will then be able to use the provided information to perform a necropsy and send feedback to the field personnel. This project is intended to support remote field diagnostics in veterinary medicine.

## CONTENTS

<b>1</b>	<b>Project Purpose and Goals</b>	<b>2</b>
<b>2</b>	<b>Progress and Plans</b>	<b>2</b>
2.1	Completed Parts of Project . . . . .	2
2.1.1	Database . . . . .	2
2.1.2	Android . . . . .	2
2.1.3	Website . . . . .	3
2.2	Parts of Project Left to Complete . . . . .	3
2.2.1	Database . . . . .	3
2.2.2	Android . . . . .	3
2.2.3	Website . . . . .	3
<b>3</b>	<b>Problems</b>	<b>4</b>
<b>4</b>	<b>Code and Images</b>	<b>4</b>

## 1 PROJECT PURPOSE AND GOALS

The completed project components are intended to serve as a means of communication between personnel in the field and pathologists in the laboratory. Its purpose is to improve a team's ability to perform remote diagnostics by providing a convenient way for teams to communicate information. The OVDL wants a native Android mobile application that collects field data and images, stores the information on a native SQLite database, sends them to the labs MySQL database, and gets real time feedback from the lab. The lab will interact with the database and user's submissions through the website.

## 2 PROGRESS AND PLANS

### 2.1 Completed Parts of Project

#### 2.1.1 Database

Figure 1 displays the code used to create a Submission table in the Mysql database. Currently we are only creating a small version of the Submission table for testing purposes. Once we have determined we can reliably send information from the SQLite database to the server where Mysql is located then we would begin the process of fully flushing out the database on the server end. Figure 2 displays the current SQLite Database in android. This database is more flushed out in order to test portions of the android app. Like the MySQL database, the tables are currently in testing form and as such do not have the all the necessary functions needed for each table. As an example we do not have a Accessor for the column Submitted. Though we do have them for the internalID, Title, and Date Of Creation.

#### 2.1.2 Android

The team has made significant progress on the Android application. The home screen has been completely implemented. All of the navigation buttons have been added and they all take the user to the corresponding screens. Additionally, the toolbar has been implemented and can take the user to any of the relevant screens.

Clicking on the "Create Account" button successfully opens up a web browser. For now, it just takes the user to the Oregon Veterinary Diagnostic Laboratory's website.

For the create submissions screen, all of the input fields were added. The "Save Draft" and "Submit" buttons correctly store some of the submission data into the SQLite database. Additionally, the "Add Pictures" screen has been implemented. Users can upload five pictures using either the camera or the phone's gallery and can also delete the pictures they uploaded. Also, the application can track all of the meta data of the pictures, including the name of the picture, the date it was taken, and the GPS coordinates.

For the "View Drafts" screen, all of the drafts stored in the database can be seen in a list. Additionally, users can remove drafts from the database by long clicking on the entry in the list.

For the "View Submissions" screen, the user can view all of the submissions that they submitted to the database in a list. The user can also click on an entry and the app will successfully navigate to that submission's detailed page. Like the View Drafts screen, the user can delete submissions from the database by long clicking on the submission in the list. Each entry in the list also shows the relevant information such as the submission title, the case ID, and when the submission was created.

### 2.1.3 Website

All the pages of the website have been created with basic layouts and some input validation for the login and registration pages. The pages ensure input fields are filled properly before submitting to the database for processing. The main page is the submissions page that has a table of all the submissions available to the user. The table shows the submission title, the name of the user that submitted it, its case ID and the date it was received. When clicked the individual submissions open a case detail page with the rest of the submission data. This is where messages will be processed. Users will be able to send messages about the submission to the submitter's phone.

There is also an account page where users can change their username or password. An about page is also included to provide users with information about the site, the app, and how to use them or contact people who can assist them. The pages are all accessible to each other through a navigation bar at the top of the pages. The website is currently hosted on an OSU server, but it will be moved to the client's server when it is ready.

## 2.2 Parts of Project Left to Complete

### 2.2.1 Database

Both databases need to be fully created. Figure 3 is a diagram of the tables needed to be implemented for each database, and a key which explains the colors associated with the table. In addition to the tables needing to be added to the respective database, functions for the SQLite tables also need to be implemented in the android app. These functions include: Accessors, Modifiers, and any unique functions associated with the table.

### 2.2.2 Android

The team still needs to fully complete all work that uses the API. The client is still developing the API, so the team hasn't been able to work on any tasks that involve it. This includes handling login and logout, sending data from the phone to the server, and receiving comments on submissions from the website. Additionally, clicking on the "Create Account" currently opens up the OVDL's website. The team will need to change the URL that the app opens up once the website has been hosted. The team also still needs to store the rest of the submission data into the SQLite database and have that data be stored on the detail submission page. The drafts functionality still needs to be implemented because clicking on a draft doesn't navigate the user to a new screen with the fields filled out already. Lastly, the team needs to implement the settings and instructions screens.

### 2.2.3 Website

The website still needs to be transferred to the client's Ubuntu server. There is still work being done on the server so the web pages are being designed separately. The website has also not been connected to the database for a similar reason. The MySQL database is still being completed as well as the API that will facilitate their communication. Once those pieces are ready to connect some choices about the login and registration process will need to be made. The clients will need to decide how much security is necessary and specify the differences between user groups. We have discussed having levels of privileges for general users, OVDL affiliated users, and OVDL veterinary pathologist users. It is still unclear how security levels will be assigned and enforced.

Another important piece that is still unclear is messaging. We know pathologists will need to send messages to users about their submissions, but it is unclear if pathologists will also send messages to each other or how security levels will affect the messaging system. Unstable network connectivity is also a concern so we will need to implement a way of handling potential internet interruptions.

### 3 PROBLEMS

The first issue we had was one of our team members was unable to push anything to our android repository. He was able to pull information but when he tried to push it would say he did not have authorization. We checked the repository to see if he was added as a collaborator. That was not the issue. We met as a group to see what was wrong. We had him reinstall android studio and tried switching Github accounts. We eventually figured out that the issue was that his laptop kept autofilling one of his other Github accounts information.

An issue we had during testing the submissions table in our sqlite database was any changes we made to the sqlite submission table would not be shown in the emulator of android study. We tried using different emulators but the same error would occur. The solution was we needed to recreate the tables for the changes to occur. It did enforce our belief the database was not being created every time we opened the app though.

A permission issue we all had was connecting to the mysql database from outside the server. If we used putty, we could get to the server and interact with the tables but we could not interact with the table without using putty. We tried direct connections but we kept getting errors stating we did not have permission. The solution this time was found out by our client, who discovered we needed to create an ssh tunnel on the server in order for us to connect outside of it. Once this was created we had no more problems connecting.

When we were testing to see if we could add submissions to the sqlite database we could not figure out why any new additions could not be displayed. We used logs to see the flow of how a submission was processed in our app and determined it was being added correctly, but we simply could not retrieve the information that was stored. With the help of the logs we then determined to look deeper into the function used to retrieve the data from the sqlite server. Turns out the function had a logic error and was trying to display something that could not happen. Once we fixed the logic problem in the function, the submissions were properly displaying.

Another issue that we had was that we haven't been able to use the API yet because the client is still developing it. As such we have not been able to test the connection between the android and the database sever.

### 4 CODE AND IMAGES

```
CREATE TABLE 'SUBMISSION' (
  'master' int(11) NOT NULL,
  'Title' varchar(20) DEFAULT NULL,
  'DateOfCreation' date DEFAULT NULL,
  PRIMARY KEY ('master'))
```

**Fig. 1:** Code to create Submission Table in the Mysql Database

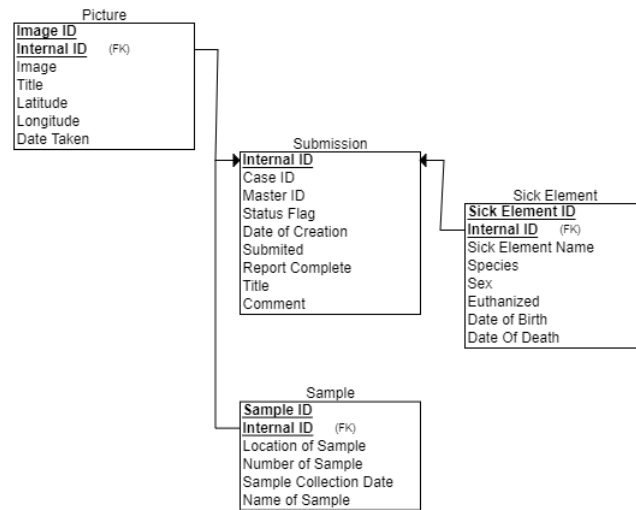


Fig. 2: SQLite Alpha Phase

• KEY:

- Red: Both MySQL and SQLite need this table added.
- Blue: Only MySQL need this table added.
- Green: Both MySQL and SQLite have this table implemented.

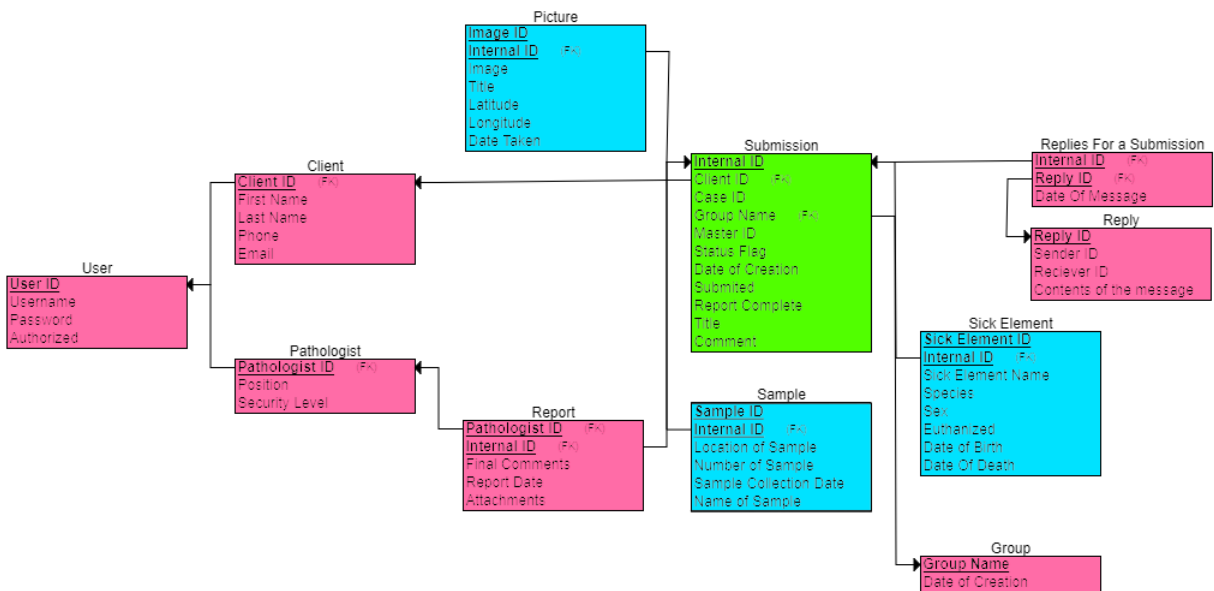


Fig. 3: Future Plans for Databases

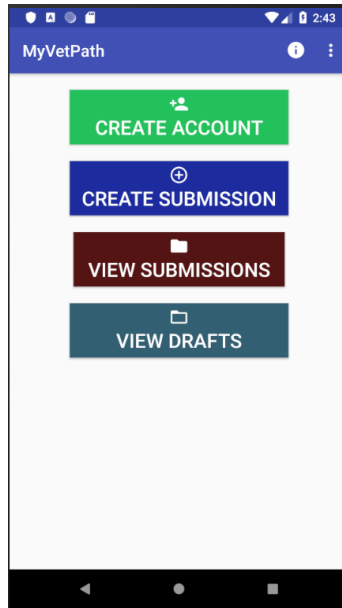


Fig. 4: Home Screen

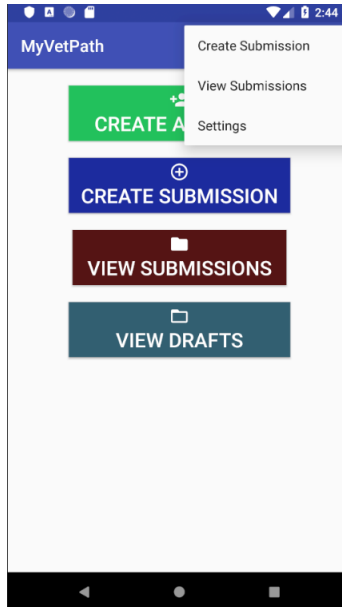


Fig. 5: Toolbar

← Create Submission ⓘ ⋮

**Submission Title**  
Title

**Group Name**  
Group Name

**Location of Sample**  
Location

**Number of Samples**  
Number of Samples

**Name of Samples**  
Sample Name

**Sample Collection Date**  
📅

**Sick Element Name**  
Sick Element Name

**Species**  
Species

**Sex:** ▾

Euthanized ☐

**Date of Birth**  
📅

Fig. 6: Create Submissions Screen

← Create Submission ⓘ ⋮

**Name of Samples**  
Sample Name

**Sample Collection Date**  
📅

**Sick Element Name**  
Sick Element Name

**Species**  
Species

**Sex:** Male ▾

Euthanized ☐

**Date of Birth**  
📅

**Date of Death**  
📅

**Comment**  
Comment

📷

SAVE DRAFT

SUBMIT

Fig. 7: Create Submissions Screen Part 2)



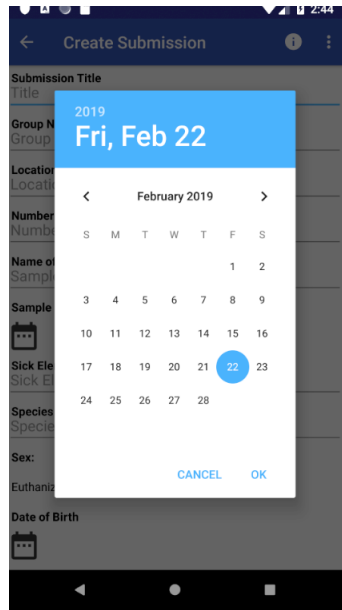


Fig. 8: Date Picker

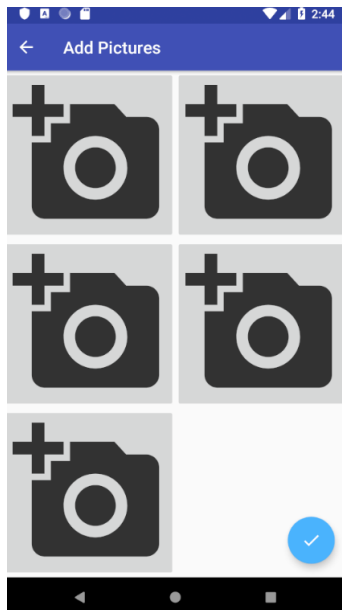
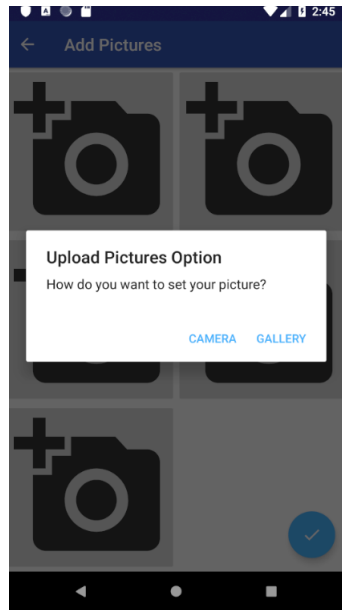
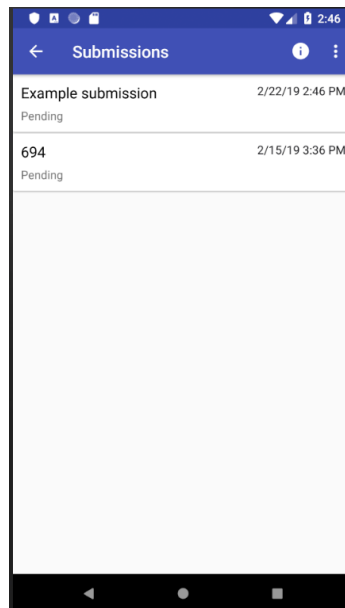


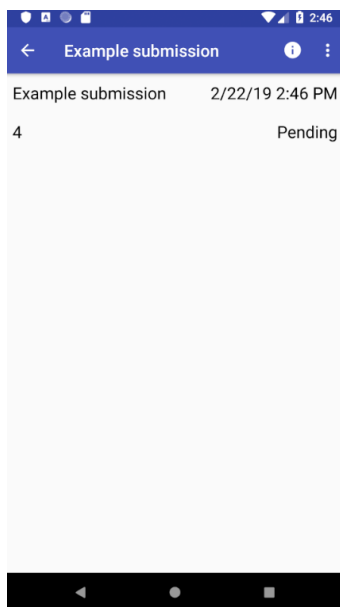
Fig. 9: Add Pictures Screen



**Fig. 10:** Add Pictures Prompt



**Fig. 11:** View Submissions Screen



**Fig. 12:** Detailed View Submissions Screen

## Oregon Veterinary Diagnostics Laboratory

Portal for the OVDL remote diagnostics app

Email address:

Password:

☐ Remember me

Login

Don't have an account?

Register New Account

**Fig. 13:** Website Login Screen

## Register New Account

Create a new account to send remote reports to the lab.

Email address:

Password:

Confirm Password:

Position:

Register

**Fig. 14:** Website Registration Screen