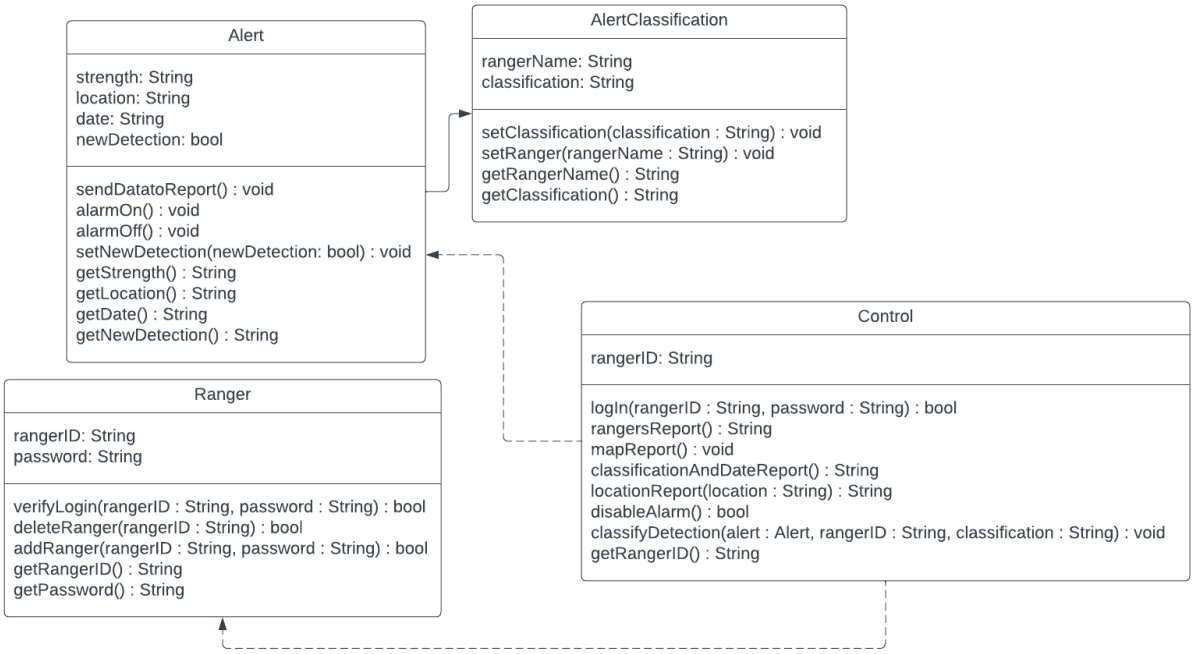

Description of Classes

Alert

The Alert class is designed to retrieve information from the third-party sensor that detects animal noises. This class includes the functionality for sending data to generate reports as well as controlling the alarm that goes off when a detection is received from the sensors.

AlertClassification

The AlertClassification class is an extension of the Alert class which adds variables for classification and the name of the ranger that reviewed the detection. This completes the information necessary for a detection to be included in a report, so each Alert should have an associated AlertClassification.

Ranger

The Ranger class represents all of the individual ranger profiles stored in the User Database depicted in the Software Architecture Diagram. This class will mostly be used to verify the rangers when they log in.

Control

The Control class includes the functions the rangers can perform on their computer. This class facilitates the user storing information related to the detections and retrieving reports.

Description of attributes

The Alert class has four attributes. The first attribute, animalNoise, is a String that defines the kind of animal that made the sound detected. The strength String attribute measures the intensity of the sound. The location String attribute is defined by the location of the sensor when it detects a sound. The date String attribute is determined by the date when the sound is detected. Finally, the newDetection attribute is a boolean variable that is set to true whenever a new detection is received from the Animals-R-Here system. The AlertClassification class builds onto the Alert class with two additional attributes, one for the classification of the detection set by the ranger and one for the ranger that reviewed the detection, both of which are String type. The Ranger and Control classes both use a String attribute for rangerID that is unique to each ranger. The Ranger class contains an additional String attribute for password, which will be used alongside rangerID when the user logs in.

Description of operations

The Alert class has seven functions. The first is sendDataToReport() which returns the information of a given detection to be included in a report. The class also has two operations associated with the alarm system, alarmOn() and alarmOff(). The alarmOn() function is triggered whenever the newDetection bool variable is true. The alarm will turn on and stay on until a ranger manually turns it off using the alarmOff() operation. The last 4 functions, getStrength, getLocation, getDate, and getNewDetection are all getter methods that return the current state of that attribute. The AlertClassification class has four methods. The first is setClassification() which allows the ranger currently at the computer to review the detection and classify it as “definite”, “suspected”, or “false” according to how likely it was to be an actual mountain lion detection. The setRanger() function will associate the current ranger with the detection so it can be shown in the reports. The final two functions are getRangerName() and getClassification() are getter methods for the rangerName and classification attributes. The Control class has eight functions. logIn(String rangerID, String password) is used in the Control class to allow a ranger access to the database so they can request reports. classifyDetection(Alert alert) is used to select an alert to classify. There are four functions within the Control class that generate reports. The classificationAndDateReport() function retrieves a report that includes all mountain lion detections organized by date and classification. The locationReport(String location) takes one String argument indicating location and returns to the user a report including all mountain lion detections at a specific location. The requestMap() function provides a map of the park and the surrounding area indicating all stored detections. The rangerReport(string ranger) function takes one String argument indicating a specific ranger and returns to the user all detections reviewed by that ranger. Elsewhere in the class, disableAlarm() is used to stop the alarm from making

sound. Finally getRangerID() returns the current state of the rangerID attribute which for our purposes will be the rangerID for the ranger that is currently logged in. The Ranger class includes three functions managing a ranger's ability to use the system. deleteRanger(String rangerId) is used to stop allowing a particular ranger to access the database. The second function, addRanger(String rangerID, String password) is used to create a new Ranger in the system. Finally, verifyLogIn(String rangerID, String password) checks to ensure that a ranger's ID and password match. All three of these functions return a boolean value indicating success (true) or failure (false). The final two functions for the Ranger class are getters for the rangerID() and password() attributes that will be used during the login process.