Mountain Lion Detection System

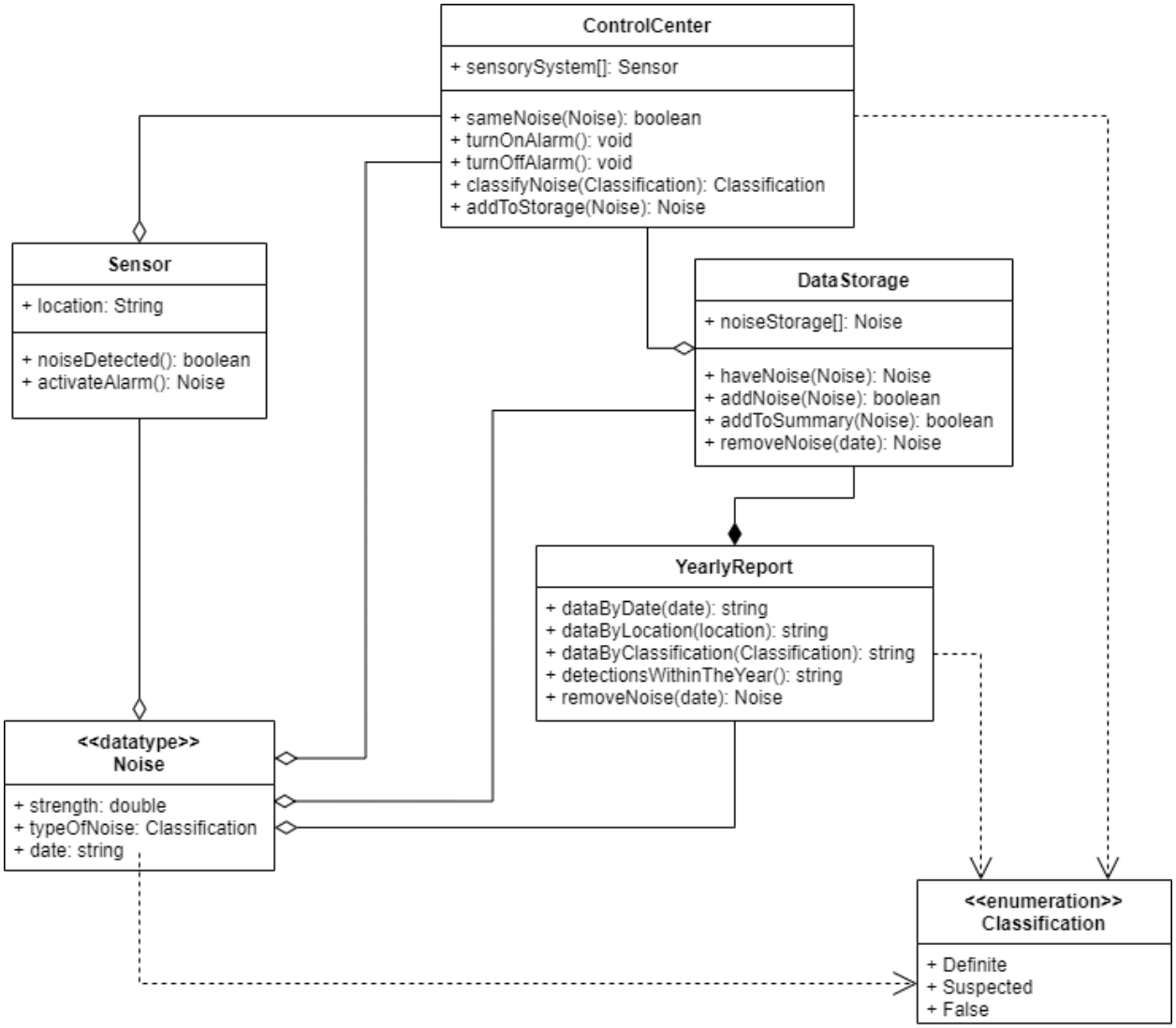
Prepared by: Jonathan Trinh, Christopher Diep, Jasmine Nguyen, Simon Rofaeel

October 15, 2021

System Description

The Mountain Lion Detection System belongs to the San Diego County Parks and Recreation Department and is based on the development of the animal detection system by the Animals-R-Here company. The purpose of the system is to detect all mountain lions and other types of animals found around the parks. Within the system, there are noise detection sensors in a five square mile radius, as well as a detection classifier for the various types of animal noises. There are controlling computers located in the park ranger stations and all park rangers have access to them. Within the system, alert messages can be requested to be sent to the controlling computers with the strength, location, and type of noise that is detected. The control program on the controlling computers will also sound an alarm whenever there are alerts received within the animal detection system.

Architectural Diagram



Class Description

ControlCenter: The controlCenter class provides functions for the user to access parameters and states, and handles a number of conditions that may occur. The controlCenter class inherits from all of the other classes. This information updates when the backend data is changed.
Attributes: sameNoise(Noise), classifyNoise(Classification)
Operations: sensorySystem[], turnOnAlarm(), turnOffAlarm(), addToStorage(Noise).

Sensor: Alarms are only activated when the sensor detects a noise classified as a mountain lion within the range of the detector. The sensor class has an association to the controlCenter class and noise class, therefore it will send an implementation of activateAlarm() when a noise is detected.
Attributes: location, noiseDetected()
Operations: activateAlarm()

Noise: The noise class is a data type class that holds a collection of attributes shared between all classes and abstracts the operation of the sensor. It is an aggregation to dataStorage class, yearlyReport class, and controlCenter class.
Attributes: strength, typeOfNoise, data

DataStorage: This class holds data about different types of noises classified by the user/ranger in the controlCenter class to reduce error and aggregates from the controlCenter class.
Attributes: haveNoise(Noise)
Operations: noiseStorage[], addNoise(Noise), addToSummary(Noise), removeNoise(data)

YearlyReport: The yearlyReport class holds properties from the noise class, sensor class, and classification class, and is a composition of the dataStorage class as it cannot exist meaningfully without it. This class allows the user to request several reports.
Attributes: dataByDate(date), dataByLocation(location), dataByClassification(Classification), detectionsWithinTheYear()
Operations: removeNoise(date)

Classification: An enum class that contains three constants, “Definite”, “Suspected”, and “False”. The returned enum object works as an iterator for the activateAlarm() method as it can sequentially fetch elements stored in the typeOfNoise object. The controlCenter class, yearlyReport class, and Noise class are dependent on the classification class.

Attributes and Operations

Control Center

Attribute / Operation	Type	Description
sensorySystem[]	Sensor	The system employs a sensor to indicate whether or not an animal is detected.
sameNoise(Noise)	Boolean	The system compares newly detected noises against animal detection classifications to determine if new noise has been detected before; returns true if the same noise is detected and false otherwise.
turnOnAlarm()	Void	The system shall turn on the alarm if noise is detected and return the control to the caller.
turnOffAlarm()	Void	The system shall turn off the alarm when prompted by the user.
classifyNoise(Classification)	Classification	The user shall classify the detected noise-type.
addToStorage(Noise)	Noise	The system shall add newly detected noise to the system storage.

Sensor

Attribute / Operation	Type	Description
location	String	The system shall display the location of the detected noises.
noiseDetected()	Boolean	If a noise is detected, sensors will alert the system; returns true if noise is detected and false otherwise.
activateAlarm()	Noise	If a noise is detected, the system will activate the alarm; returns true if the alarm is activated and false otherwise.

<<datatype>> Noise

Attribute / Operation	Type	Description
strength	double	The system shall report the detected noise strength level in terms of double.
typeOfNoise	Classification	The system shall classify the type of noise detected.
date	String	The system shall display the date of the detected noise.

DataStorage

Attribute / Operation	Type	Description
noiseStorage[]	Noise	The system shall have a declared noise storage.
haveNoise(Noise)	Noise	The system determines if a noise is detected.
addNoise(Noise)	Boolean	The system shall add detected noises to the system's storage; return true if noise data is added to the system and false otherwise.
addToSummary(Noise)	Boolean	The system shall add detected noises to the system's storage summary; return true if noise data is added to the system summary and false otherwise.
removeNoise(date)	Noise	The system shall remove detected noises older than 30 days from the system's storage summary.

YearlyReport

Attribute / Operation	Type	Description
dataByDate(date)	String	The system shall allow for yearly reports, filtered by date of noise detection and data logging.
dataByLocation(location)	String	The system shall allow for yearly reports, filtered by location of noise detection and data logging.
dataByClassification(Classification)	String	The system shall allow for yearly reports, filtered by classification of noise detection and data logging.
detectionsWithinTheYear()	String	The system shall report the noise detections received within the year.
removeNoise(date)	Noise	The system shall remove noise detection data from the yearly report after the data has surpassed a year.

<<enumeration>> Classification

Attribute / Operation	Type	Description
Definite	N/A	Within the constructor of various attributes, if a noise is definitely detected, the system will classify as “Definite”
Suspected	N/A	Within the constructor of various attributes, if a noise is suspected, the system will classify as “Suspected”
False	N/A	Within the constructor of various attributes, if a noise is not detected, the system will classify as “False”

Development Plan and Timeline

Members	**Roles**	**Timeline**
Christopher Diep	Responsible for making architectural diagram of all major components	5 days
Jasmine Nguyen	Responsible for making class descriptions	2 days
Jonathan Trinh	Responsible for attributes and operations description	3 days
Simon Rofaeel	Responsible for system and operations description	2 days