San Diego County Parks and Recreation Department Mountain Lion Detection System Requirements Specification

Prepared by: Gabriela Calvo

February 17, 2023

Introduction and Overview

Our mountain lion detection system is intended to be used by the San Diego County Parks and Recreation Department for population safety, research, and preservation efforts. The first section of this document specifies user requirements by describing the overall context and intent for our system. The following section discusses system requirements; detailing our use cases (functional requirements) and overall quality expectations (not-functional requirements) for the system. The final section contains additional considerations concerning future expansion of the system.

User Requirements

Our system is a control program intended to be used by San Diego Park Rangers strictly on computers already at ranger stations. It will serve as an interface for making use of the data collected by our third party animal detection system. The main functions of this control program from a user perspective will be responding to detection alerts, classifying detections, and generating reports of previously collected data. Because of storage concerns, our system will only store complete alerts from the last 30 days. There will be a summary generated for alerts within the past year that are older than 30 days. The third party detection system we are using are sensors made by Animals-R-Here.

The Animals-R-Here sensors come pre-programmed with the ability to:

1. Place noise detection sensors within 5 square miles
2. Select the type of animal noise to detect

3. Set up alert messages to be sent to a controlling computer (to interact with our system) based on type and strength of the noise detected
4. Send alert messages containing type, strength, and location (accurate to 3 meters) of the detection

The actions offered to the user are of 3 main categories;

1. **Handling alert messages**

   Alert messages will cause an alarm to go off that requires a user to manually turn it off. The alarm will not sound again unless another detection is made.

2. **Classifying detections**

   Each alert message will have to be reviewed by a ranger and classified based on probability of an actual mountain lion setting off the sensor rather than a false flag. The classification options are definite, suspected, and false.

3. **Requesting reports**

   The user will be able to request 4 types of reports:

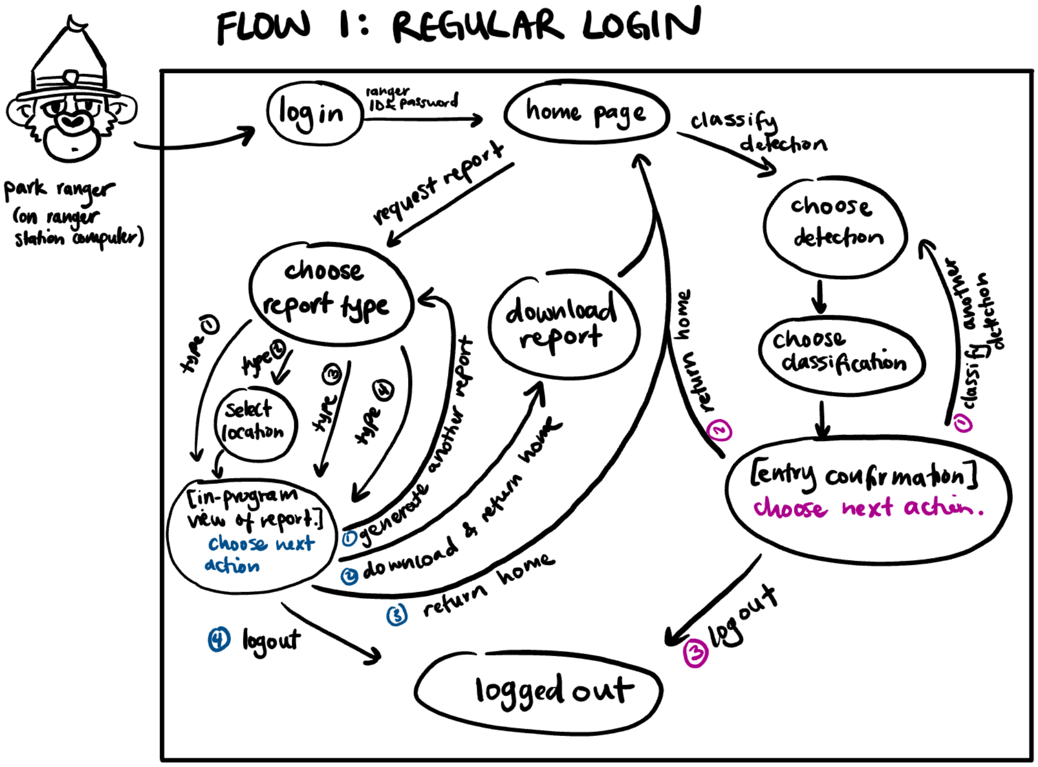
   1. Report of all detections organized by date detected and classification
   2. Report of all detections at a specified location
   3. Map of the park indicating all detections within the park extending 2 miles into the surrounding area
   4. Report of detection classifications organized by ranger

System Requirements

Functional Requirements

Below are use cases modeling intended interaction with the system by a park ranger. Park ranger is the only possible type of user, from which two alternate flows (Use Cases 1&2) have been generated. Within this system, regular login means that the park ranger is logging in for general upkeep rather than in response to an alarm.

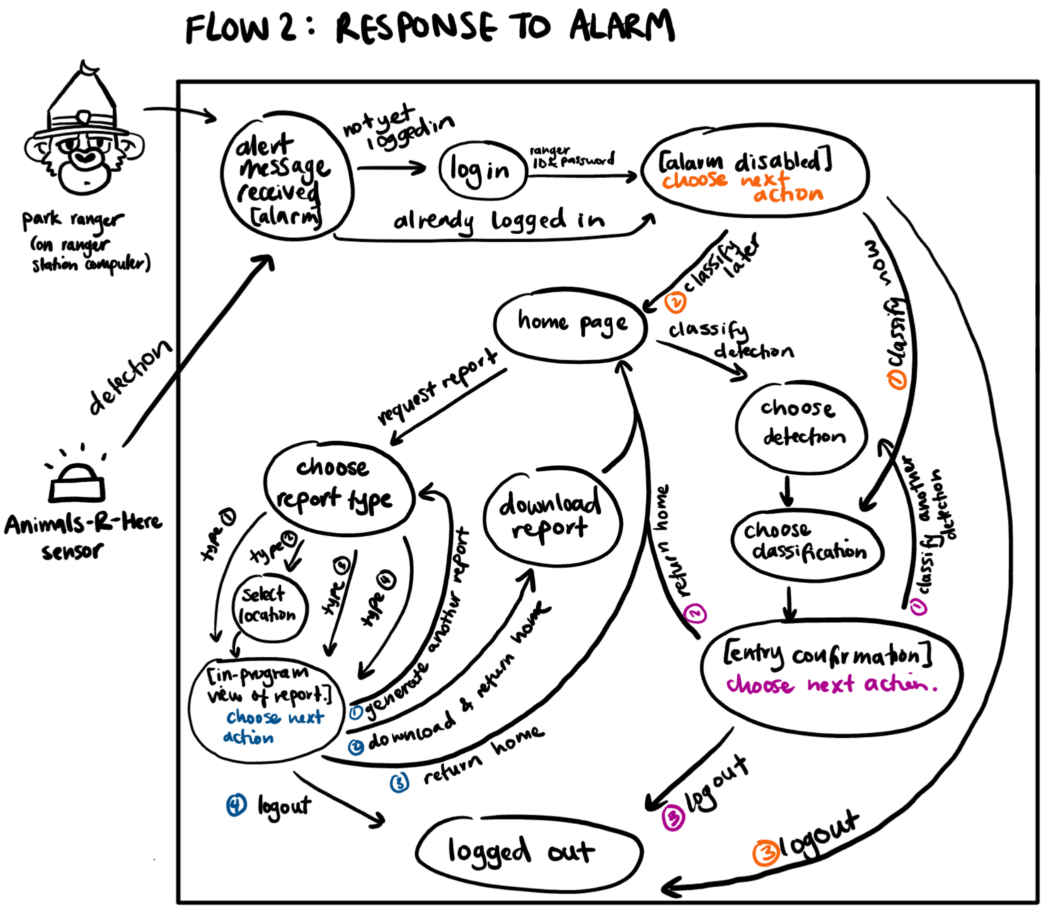
Use Case 1:



For this use case, the user begins as the login page. After entering their ranger ID and password, they are given the homepage options to request a report or classify a detection. The functionality of these two main branches are as follows;

1. If the user chooses to request a report then they are prompted to choose which type of report. The types of reports are as indicated in the user requirements section; Type 2 is the only report that requires an additional input for location before generating the report. The user is then shown an in-program view of the report and given four options; generate another report (takes the user back to choosing report type), download and return home (downloads report to station computer), return home, or logout.
2. If the user chooses to classify a detection, they are prompted to choose a detection (ordered by date), then to choose a classification (definite, suspected, or false). They are given confirmation that the detection was updated and given three options; classify another detection (takes the user back to choosing a detection), return home, or logout.

Use Case 2:



Majority of the interactions for Use Case 2 will be the same as Use Case 1, the main difference is the starting point. The first interaction for this use case is an alert message triggering the alarm within the control system. From there, the system verifies whether or not the user is already logged in. If they are, the alarm is disabled by the user choosing their next action. If not, the user is prompted to log in, then the alarm is disabled by the user choosing their next action. From here there are three options provided that all converge with Use Case 1;

1. Classify detection immediately
    a. This converges with Use Case 1 at the stage where the user is prompted to choose a classification for the detection.

2. Classify the detection later
    a. This assumes that the user wants to remain logged in and converges with Use Case 1 at the homepage, storing the detection within the system under the assumption that it will get classified at a later time.
3. Logout
    a. This converges with the ending stage of Use Case 1, logging out the user and storing the detection within the system under the assumption that it will get classified at a later time.

Non-functional Requirements

The most important qualities for our system (in no particular order) are correctness, learnability, interoperability, and usability.

Correctness

Since the system is intended to collect data and aid in research, it’s important that data is being stored correctly. This will require that the system has reliable memory that can handle storing large amounts of data. As detailed in the User Requirements, we have placed a constraint on how long data is stored in order to ensure that our system won’t get overwhelmed.

Learnability

A significant portion of the data collected is specified, entered, or checked by a park ranger so it's important that the features and capabilities of the system be easy to learn. This will be applied to our system by keeping the functionality straightforward and simple. The exact interactions between our control system and the third party detection system do not need to be specified to the user.

Interoperability

All of the data collected is going to come from third-party sensors, so it is essential that our control program has consistent exchange and interaction with the Animals-R-Here detection system. This is a reasonable quality to achieve since we know that our system will only be interacting with one other system.

Usability

Dependably carrying out our system's purpose is especially relevant given the specificity of our context. Since our system is only concerned with one type of user, one other operating system, and three main operations, ensuring that those operations can be carried out effectively, reliably, and satisfactorily should be at the center of our development process. This will involve thorough testing and consideration for edge cases.

Other

In addition to the considerations detailed in previous sections, our system has the potential to be used across the California State Parks system in the future. This should be kept in mind during our development process by forming our system in a way that is easy to modify for use in other parks within California.