

**Lab 1- Pest Patrol Product Description**

Lukas McKennedy

Department of Computer Science, Old Dominion University

CS 411W: Professional Workforce Development II

Professor James Brunelle

September 18, 2022

### **Table of Contents**

1 Introduction.....	3
2 Product Description.....	4
2.1 Key Features and Capabilities.....	4
2.2 Major Functional Components (Hardware/Software).....	5
3 Identification of Case Study.....	7
4 Prototype Description.....	7
4.1 Architecture (Hardware/Software).....	8
4.2 Features and Capabilities.....	9
4.3 Developmental Challenges.....	11
5 Glossary.....	13
6 References.....	14

### **List of Figures**

Figure 1: Pest Patrol Major Functional Components Diagram.....	6
Figure 2: Pest Patrol Prototype Major Functional Components Diagram.....	8

### **List of Tables**

Table 1: Pest Patrol Features Table.....	10
--	----

## **1 Introduction**

Run-ins with pests are common in most communities; however, these experiences are often isolated and means of reporting are scarce and not centralized, so there is rarely community awareness of, or an informed collective response to, these encounters. There is no reliable way for members of a given community to stay informed on local pest encounters; due to the lack of real-time awareness of pest presence and location, there are many encounters that could be prevented. Even though many people have pest infestations and learn from experience how best to deal with them, the guidance that they could give to people who are dealing with these problems for the first time is not consolidated or is inaccessible. The lack of awareness of current pest infestations in an area can make it difficult to effectively plan for outdoor activities and be able to mitigate pest-related problems; for example, if someone is planning fireworks at night, but does not know that an area has recently become infested with mosquitos, they may not bring the proper mosquito repellant, which could cause them to be miserable.

A solution to the problem of unawareness of local pest problems is Pest Patrol. Pest Patrol is an application designed to protect communities from pests by making it possible to proactively respond to their presence. The application works by keeping users informed of all pest encounters in their area based on reports from other users, as well as allowing communities to track outbreaks with customizable heat mapping. It also aggregates knowledge and experience gained by people who have encountered pests in the past and allows that knowledge to be utilized by community members in future encounters. As a result, users can safely plan outdoor activities with access to up-to-date information about pest problems at specific locations.

## **2 Product Description**

Pest Patrol is a web-based, cross-platform supported application that uses community crowdsourcing to provide real-time awareness of pest issues and locations, contains a knowledgebase of past incidents and strategies, allows users to directly offer suggestions to other community members on how to deal with pest problems, and uses heatmapping and predictive modeling to keep members of a community informed of pest-related situations in that community. Its goal is to make communities safer by minimizing the frequency of unwanted pest encounters as well as to diminish the severity of the pest encounters that do occur.

### **2.1 Key Features and Capabilities**

The objectives of Pest Patrol are to provide a streamlined interface for reporting and educating on pest encounters in a community, enable users to tag reported incidents with their exact location, provide a means for users to communicate about their encountered incidents, and consolidate reported incidents and related discussions. The key features and capabilities that Pest Patrol provides in support of this objective include: the user's account Dashboard, Incident Map, Pest Alerts, Thread Activity and Recent Activity, Direct Messaging, and the Community feature. The application also allows customization of the profile through Profile Settings, as well as the ability to customize a search with the Search Options function.

Pest Patrol can be used in one of two ways: it can be used in view-only mode without registering, or an account can be created using a valid email address. Registered accounts can be customized in the Profile Settings, allowing for input of user information and photo, updating of login credentials, and customization of incident display and alert settings. The account Dashboard is the first page that is loaded when logging in with an account; it provides access to all of the other features in the application. It has three view modes, including the Incident Map

mode which displays the Incident Map, the Discussions mode which shows a list of discussion threads related to reported incidents, and the Hybrid mode which simultaneously displays the Incident Map, the Discussion threads, and the currently selected thread. The Incident Map lists reported incidents based on the location of the user; the incidents are customizable based on the location, age of the incident, type of pest, and user. The Incident Map allows users to report new incidents on the map, and it also displays heatmapping with alternate map views based on the age of pest activity and the type of pest. The Pest Alert function allows the user to receive either SMS or email alerts about recent incidents in the vicinity of their location. The type of alert can be changed in the Profile Settings.

The Thread Activity module of the application displays activity related to threads that the user has either created or in some way participated in. It provides quick and easy access to relevant discussions, and it also allows users to subscribe to threads they want to monitor. The Recent Activity module includes all recent events in a user's community such as new incidents and new threads or thread activity related to the community. The Recent Activity module also includes anticipatory alerts about potential incidents based on machine learning analysis of historical data. The application's Direct Messaging feature allows communication between users without needing to use discussion threads, and the Community aspect lets users follow or friend each other, as well as report inappropriate behavior or use of the application. The Search Options are an ad hoc way of filtering items on the incident map without needing to create specific customized settings.

## **2.2 Major Functional Components (Hardware/Software)**

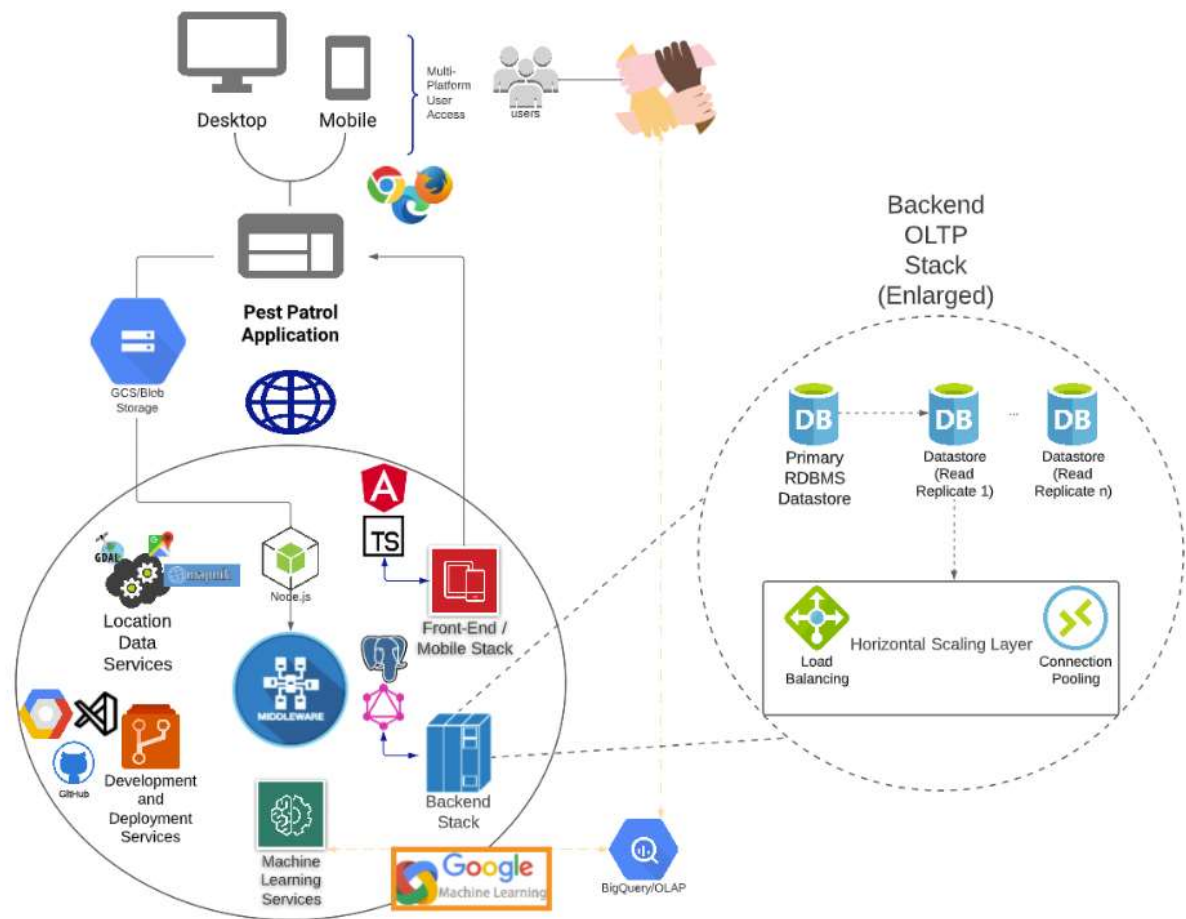
**Figure 1***Pest Patrol Major Functional Components Diagram*

Figure 1 above is the Major Functional Components Diagram for the Pest Patrol application. The application can either be utilized via desktop or mobile device. If used on a mobile device, the mobile application can be used; otherwise, it can be accessed through a web browser on any mobile or desktop device that has internet access, camera access, and a keyboard and mouse or touch screen. The front end of Pest Patrol is developed in JavaScript using Angular with TypeScript; the back end Online Transactional Processing (OLTP) stack is built with PostgreSQL and Google Cloud SQL, while Google Cloud Storage is used for blob and image

data. Node.JS is used for the middleware, which also includes the Location Data Services, as well as the Google Machine Learning Services. The code is maintained in a GitHub repository and utilizes CI/CD services. The development is done through Microsoft Visual Studio Code.

### **3 Identification of Case Study**

There will be multiple case studies for Pest Patrol. They will consist of individual members of a community, people planning and executing a camping trip, outdoor businesses, and large communities or cities. Individual community members will look for and report pests within their communities, and view alerts of other incidents reported within the communities. People planning a camping trip will view the map of their destination, look for areas with pest sightings, and determine what they need to deal with or avoid the pests. Outdoor businesses will check for and report sightings of hazardous pests in the business area. Cities will create plans to address large pest problems in their communities and implement them using the application as a guide.

Individuals in communities, those planning outdoor trips, outdoor businesses, and cities are customers for Pest Patrol; however, there are some other stakeholders as well. These include pest control companies, homeowner associations, government agencies such as the United States Fish and Wildlife Service, and researchers who are interested in the data for pest population tracking or studies.

### **4 Prototype Description**

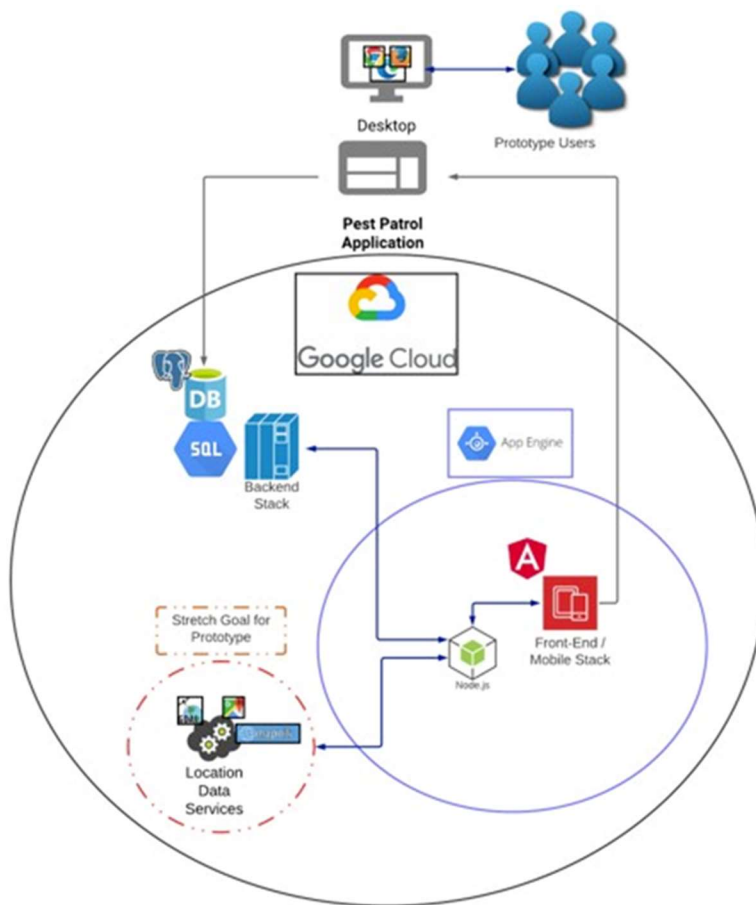
Pest Patrol's prototype will have limited features due to time constraints. In order to demonstrate satisfactory proof of concept, only the web application will be implemented in the prototype; however, the web application will still be accessible via mobile device as well as desktop.

#### 4.1 Prototype Architecture (Hardware/Software)

Figure 2 (below) shows the Major Functional Components Diagram (MFCD) in the context of the prototype.

**Figure 2**

*Pest Patrol Prototype Major Functional Components Diagram*



The application will still be accessed by either a mobile device or desktop, but only through the web browser. The front end will still be built on Angular and will run on Google App Engine in Google Cloud. The Node.JS middleware will run on Google App Engine in Google Cloud as well. If time permits, Location Data Services will also be integrated with the middleware. The



back-end data of the prototype will be held in a Google Cloud SQL instance with Postgre SQL. Time permitting, there will be mock data in a Big Query Dataset in order to demonstrate AI analytic capabilities.

#### **4.2 Prototype Features and Capabilities**

The prototype for Pest Patrol will demonstrate the application's functionalities, including opening the web application, demonstrating the Dashboard and the ability to access the other modules, viewing the Incident Map and reporting, accessing Discussion threads, viewing Pest Alerts, and the Community feature (listed below in Table 1). Through this process, it will demonstrate solving the problem that communities have with staying up-to-date and sharing information on pest incidents in a given area.

**Table 1***Pest Patrol Features Table*

Function	Real World	Prototype	Current Plan
<b>General</b>			
Web and mobile compatibility	Fully Functional	Partially Functional	Partially Functional
Dashboard	Fully Functional	Fully Functional	Fully Functional
Hybrid Mode	Fully Functional	Eliminated	Eliminated
Authentication and Identification	Fully Functional	Eliminated	Eliminated
Password Recovery	Fully Functional	Eliminated	Eliminated
<b>Incident Map</b>			
Incident Map	Fully Functional	Fully Functional	Fully Functional
Incident Reporting	Fully Functional	Partially Functional	Partially Functional
Ad hoc Incident Filtering	Fully Functional	Fully Functional	Fully Functional
Heat Mapping	Fully Functional	Partially Functional	Partially Functional
<b>Discussion View</b>			
Discussion Thread View	Fully Functional	Fully Functional	Fully Functional
Expanded discussion view	Fully Functional	Fully Functional	Fully Functional
Follow/Subscribe to discussion thread	Fully Functional	Fully Functional	Fully Functional
Discussion thread creation	Fully Functional	Fully Functional	Fully Functional
Reply to discussion thread	Fully Functional	Fully Functional	Fully Functional
Provide positive/negative feedback to threads	Fully Functional	Fully Functional	Fully Functional
<b>Pest Alerts</b>			
Pest Alerts	Fully Functional	Partially Functional	Partially Functional
Alert customization	Fully Functional	Partially Functional	Partially Functional
<b>Community</b>			
Search for user	Fully Functional	Fully Functional	Fully Functional
Add friends	Fully Functional	Fully Functional	Fully Functional
Report Users	Fully Functional	Fully Functional	Fully Functional
User reputation system	Fully Functional	Eliminated	Eliminated
Automated Moderation (ML)	Fully Functional	Eliminated	Eliminated
Hide flagged content	Fully Functional	Fully Functional	Fully Functional
Account suspension	Fully Functional	Fully Functional	Fully Functional
Flag inappropriate content	Fully Functional	Fully Functional	Fully Functional
Content removal	Fully Functional	Fully Functional	Fully Functional
View flagged content	Fully Functional	Fully Functional	Fully Functional
Block user	Fully Functional	Fully Functional	Fully Functional
Content search	Fully Functional	Fully Functional	Fully Functional
Recent Neighborhood Activity	Fully Functional	Fully Functional	Fully Functional
Direct Messaging	Fully Functional	Fully Functional	Fully Functional
New thread activity notification	Fully Functional	Fully Functional	Fully Functional
New direct message activity notification	Fully Functional	Fully Functional	Fully Functional
New incident notification	Fully Functional	Fully Functional	Fully Functional
AI generated notifications (ML)	Fully Functional	Eliminated	Eliminated
Notification customization	Fully Functional	Fully Functional	Fully Functional
Predictive Modeling (ML)	Fully Functional	Eliminated	Eliminated

The Dashboard will be fully implemented, but Hybrid Mode, Authentication and Identification, and Password Recovery will not be implemented in the prototype, due to not implementing account features. The Incident Map itself will be fully implemented along with Ad

Hoc Incident Filtering, but the Incident Reporting and Heat Mapping will only be partially implemented, just enough to be able to demonstrate proof of concept. The Discussion View will be completely included in the prototype, including the Discussion Thread View and ability to expand as well as the abilities to follow, create, reply to, and provide feedback on discussion threads. Pest Alerts will also only be partially implemented. Most of the Community Function will be usable in the prototype, including the abilities to search for, friend, and report users. The ability to deal with inappropriate content will also be implemented, such as being able to hide content, suspend accounts, flag or remove inappropriate content, and block users. The prototype will allow for searching for content, looking at recent neighborhood activity, and direct messaging. Notifications will be implemented for New Thread Activity, New Direct Message Activity, and New Incidents. AI Generated Notifications and Predictive Modeling will not be implemented, and neither will the user reputation system or machine-learning-based Automated Moderation.

### **4.3 Prototype Development Challenges**

The development of this prototype will come with many challenges. There will be technical challenges due to unfamiliarity with the many different platforms and frameworks that will be used. Inexperience with tools such as Angular and JavaScript on the front end, Node.JS for middleware, or PostgreSQL on the backend can cause challenges. This project has several moving pieces that need to work together perfectly in order for the application to work correctly, which could prove challenging. The use of a cloud platform such as Google Cloud also adds another layer of complexity to the project, as well as the use of containerization through Docker.

Having so many team members is a challenge on its own because it can be difficult for everyone to have something substantial to do; each team member will have to go out of their way

to try to find something to do, instead of waiting for something to come to them. The roles that each team member was assigned do not necessarily correlate to specific tasks in the project, which could make designating specific taskings more difficult. The number of people will also make using good Git practices paramount, because it would be easy for someone to accidentally break the entire project when committing or merging their checked-out repository branch.

## 5 Glossary

**Bot Moderation:** The automatic screening of user content to ensure proper user behavior.

**Community:** The people with common interests living in a particular area broadly the area itself.

**Community Member:** An individual member of a community.

**Geo-tagging:** The process of appending geographic coordinates based on the location of a mobile device.

**Geo-targeting:** Method of determining the geolocation of an application user and delivering different content to that visitor based on their location.

**Incident:** An occurrence or sighting of a pest reported by a user.

**Pest:** Any animal or plant harmful to humans or human concerns.

## 6 References

Abell, J. (n.d.). Nuisance Wildlife Encounters on the Rise.

<https://www.homestead.org/lifestyle/homesteading-life/nuisance-wildlife-encounters-on-the-rise/>

Bugg, S., Colborne, S., & Haerther, D. P. (2020, March 2020). Great Lakes Invasives: Sea Lampreys.

<https://www.sheddaquarium.org/stories/great-lakes-invasives-sea-lampreys>

Coyle, D. (2021, September 10). Not So Fast! International Biosecurity Program Succeeds in Preventing Spread of Invasive Moth. [https://entomologytoday.org/2021/09/10/international-biosecurity-](https://entomologytoday.org/2021/09/10/international-biosecurity-program-succeeds-preventing-spread-invasive-moth-lymantria-dispar-asiatica/)

[program-succeeds-preventing-spread-invasive-moth-lymantria-dispar-asiatica/](https://entomologytoday.org/2021/09/10/international-biosecurity-program-succeeds-preventing-spread-invasive-moth-lymantria-dispar-asiatica/)

Herring, D. (2012, March 6). Climate Change: Global Temperature Projections.

<https://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature-projections>

How Warmer Winters Affect Pests. (2018, January 15). <https://varmentguard.com/blog/warmer-winters-affect-pests>

Japanese Beetle Repeatedly Eradicated from California. (n.d.). <https://www2.ipm.ucanr.edu/Invasive-and-Exotic-Pests/Japanese-Beetle/>

Lyme Disease Costs Up to \$1.3 Billion Per Year to Treat, Study Finds. (2015, February 5).

<https://publichealth.jhu.edu/2015/lyme-disease-costs-more-than-one-billion-dollars-per-year-to-treat-study-finds>

Parkman, K. (2021, September 23). Pest control statistics and trends.

<https://www.consumeraffairs.com/homeowners/pest-control-statistics.html>

U.S. Census Bureau. (2021, April 21). Residents of 14 million housing units reported seeing roaches,

14.8 million saw rodents in last 12 months. <https://www.census.gov/library/stories/2021/04/how-many-american-homes-have-pests.html>