Lab 1 – Pest Patrol Product Description

Anton Rasmussen

Old Dominion University

CS 410

Professor James Brunelle

April 11, 2022

Version 1

1 Introduction

Pest encounters happen frequently and occur widely across the United States but due to their distributed nature are a problem that appears to occur randomly. If one follows neighborhood specific Facebook Pages or if one happens to be a member of their community's *Nextdoor* Application page, one is likely to encounter more than a few incidents related to pests or encounters with pests (i.e., frequent and widely occurring encounters); but, because this reporting is ad-hoc and not centralized to a "pest-specific" application, much of the encounter data gets lost in the high volume of otherwise unrelated distributed communication (i.e., seemingly random encounters). Pest encounters happen but because there is no single source to compile pest encounter data specifically, the true frequency of these encounters is all but lost.

Even in cases where pests are entirely preventable, we find that because of the lack of information regarding where pest "hot spots" are within a community, avoiding an encounter is akin to contending with and trying to manage completely random probabilities. For example, the knowledge a family in a community gains when it overcomes infestation is knowledge that is unable to easily flow to others because there is no good, central, place to share lessons-learned, and therefore does not aid in the ability for others in the community to take proactive measures. Additionally, we find that because there is no central repository of pest information for a certain community or region, planning of outdoor activities and events comes with risks that could otherwise be mitigated more easily were there some "place" to host this information.

The Pest Patrol application is designed to address the issue with aggregating pest-specific information into one easy to use application so that a community can coordinate the fight against pests. Pest Patrol plays the role of "do one thing well" in that its design is based on filling the gaps that exist in the currently distributed ad-hoc pest reporting space. While aggregating the information allows for more transparency around the ongoing "State of the Union" for pests in a certain area, the primary benefit of this application is making it possible for communities to pro-actively take measures to respond to pests, therefore taking back some determinism in an otherwise seemingly stochastic problem space.

Pest Patrol is not just an application for aggregating pest information related to a specific area but is itself tailored to act as a "community" in the sense that users are able to share about their encounters, interact with each other to obtain helpful and often crucial key details surrounding pests, as well as trigger insight into where pests may be coming from/moving to, where they might be more or less prevalent, and/or how intense the population of pests is for a given area of concern.

Ultimately the community aspect of Pest Patrol is the key driver of keeping individuals informed, minimizing the risk of pest encounters based on user reporting, projecting future pest movements/patterns, and providing an assist in and avenue for risk-mitigation when planning outdoor activities that otherwise may have resulted in a spoiled outing, a minor injury, or worse.

2 Pest Patrol Description

Pest Patrol is a cross-platform web application that relies on community-based (crowdsourced) information aggregation tied back to location services that help to map pest-related information geographically within the community. The network effect of this community-based crowdsourcing affords the ability to raise near real-time awareness through the application's reporting mechanism because as more people use the application, more areas are monitored for pests, and, ultimately, more people fall into those "monitored" areas, thus driving the incentive for even more people to use the app. As each report is associated with a location (a location that can be made to be more- or less-granular based on the user's choice), users themselves become both the source of pest encounter information as well as the consumers of related pest encounter reporting. The ultimate power of this location information is that the intensity of pest encounters can be mapped (e.g., using a heat-map) and used for predictive modeling of pest movements over an area (e.g., based on season, pest-type, etc.).

As users of Pest Patrol work through the process of addressing any kind of pest problems directly affecting them, they can share the knowledge they learn to other members of the community with the benefit of ultimately slowing or stopping the spread of pests throughout the community. As more and more data are collected related to pests and pest encounters, it will be possible to provide help in how to prevent pests as well as how to stay safe should one find themselves in a pest encounter.

The two main goals of Pest Patrol are 1) to make communities safer by broadening pest awareness in a community and thereby lessen encounters 2) to minimize the damage specific pests can do to an area by aggregating information that will help to slow their spread and predict future spread.

2.1 Key Product Features and Capabilities

The key features of Pest Patrol begin with the fact that the application is web-based and, therefore, multi-platform. Opting to implement a web-based application instead of an Android or iOS native application means that the application will remain easy to use no matter what type of a device a person will be using. Given that a major factor of Pest Patrol is the community crowdsourcing aspect, it is important that the ability to use the app is not hampered by the technology.

Setting up an account with Pest Patrol is not a pre-requisite to benefit from the app's data. However, without being a registered user, access is limited to read-only. Again, this design choice tips its hat to the idea that we want Pest Patrol to be community-based and thus easy for all community members to use (indeed that's the key-differentiator of this app versus similar apps on the market); however, because we want to maintain a healthy community, any user who has access to write data to the application for others to see (e.g., a pest report, user comments, etc.) will have to be validated and their credentials verified.

The main screens a user will interact with are the Dashboard, a landing-page interface that affords the ability to navigate to anywhere else on the site, which provides three separate modes: *Incident Map*, *Discussions*, and a *Hybrid* experience where the Incident Map and Discussions are all on the same screen. The Incident Map is where reported incidents populate and includes attributes such as location, incident timestamp, pest type, submitter, report incident, and heat map. All these attributes are self-explanatory but what's notable is that the attributes are customizable, filterable, and will adjust based on one's location settings. In addition to user-activated functions there are also community discussions that populate based on discussion threads as well as pest alerts that send notifications of pest encounters. The discussion and private messaging features of Pest Patrol give it the true community power that we feel is necessary for such an application to thrive in the marketplace. That said, the alerting and filtering features of Pest Patrol provide the community with the true power of achieving the goals of making communities safer and minimizing the damage of pests.

2.2 Major Components (Hardware/Software)

Hardware

- *Mobile Web Application* – Internet-enabled mobile device with camera
- *Desktop Web Application* – Computer with access to internet, keyboard, mouse, and camera

Software

- *Web browser* – Pest Patrol will be tested against various device sizes using the Browser's Dev tools and against at least three different browsers (Chrome, Firefox, and Edge)
- *Front End* – In order to create a responsive JavaScript experience Pest Patrol will use a framework such as Angular.js and TypeScript (a type-safe version of JavaScript, which Angular uses)
- *Middleware* – In order to write the APIs (REST endpoints), which serve up data from the backend into the UI, we will use Node.js
- *Code editor* – Given the nice UI and the power of awesome plugins, developers may want to use Microsoft Visual Studio Code; that said, Code-editing is a personal matter so this will not be mandatory
- *Repository* – All code will be pushed to a GitHub Repository and CI/CD integrations will work through GitHub commit triggers
- *GraphQL* – Eventually it may be beneficial to serve certain data (e.g., blob/images) in a nested complex data type format using NoSQL and something like GraphQL

Database

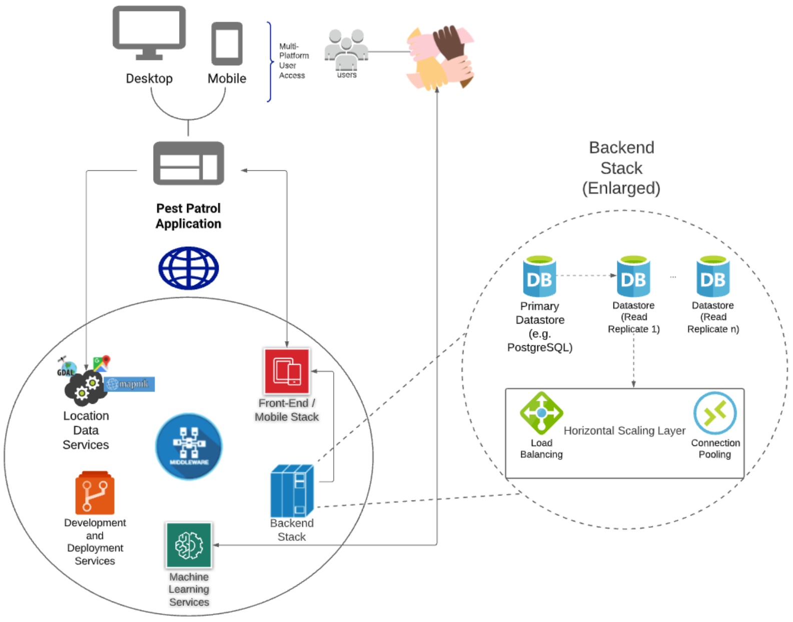
- The most likely environment for hosting Pest Patrol will be in a Cloud Environment, for this project we discussed going with Google Cloud Platform (GCP). Given that, we will use the following stack (all part of GCP):
  - PostgreSQL - for transactional data
  - Cloud Storage - for Blob Data
  - Big Query - Data Warehouse

Runtime / CICD / Microservices

- The following services are also part of Google Cloud Platform:
  - App Engine – For hosting Compute Instances
  - Cloud SQL – PostgreSQL Instances
  - Maps APIs and Services – Location Data Services
  - AI Platform – For Participant Bots and Prediction

Figure 1

Major Functional Components Diagram (MFCD)



We can begin looking at the whole MFCD from the users' point of view. Given use of a responsive front-end stack, users have multi-platform access to the Pest Patrol application through either mobile or desktop devices. Between the user accessing the application through the front-end / mobile stack and the data stored in the back-end, there exists a layer of middleware services. In the middleware we have location data services (that allow for viewing and tagging geo-location of pests), the typical front-end/back-end messaging found in web applications, machine learning services (that allow for filtering spam users/content as well as predicting likely movement of pest outbreaks over time), and CI/CD services (that allow the dev team to quickly address application fixes and improvements). By zooming in on the back-end we see that the data architecture we are employing involves emphasizing data availability through replication (this allows us to both scale up the number of users and scale out the delivery of the application across the country while at the same time maintaining a positive user experience) -- this is all assisted by a horizontal scaling layer that load-balances any writes to the database and connection pooling for caching reads based on user location or some other partitioning key.

3 Identification of Case Study

Pest Patrol is for Community Members, Hikers/Campers, Outdoor Businesses, and Cities. While the man user of Pest Patrol will likely be that same market of individuals dispersed across several apps reporting their encounters with pests, there are opportunities for people who spend a lot of time outdoors to become veritable "super" users of the app. For example, one of the major pests that people would be very interested in being alerted to (given the risks that these pests carry disease) are ticks. In certain places tick populations are so prevalent that it is almost impossible to go on a short walk in the woods and not come home with a few crawling on your clothes. In these cases, it would be both helpful and useful for hikers, campers, and other outdoor enthusiasts to have an easy way to log their encounter, including their exact location of encounter and any pictures they may have that help to identify the species of tick with which they have come in contact. In addition to being contributors of this data, other hikers and campers could be informed of the locations with the most densely distributed tick populations and thereby know to avoid those areas during their hike or when looking for a place to set up a tent. Given the value of this information it makes sense that businesses oriented toward outdoor activities would also want to both contribute to as well as consume information from Pest Patrol.

In addition to outdoor enthusiasts and the companies that serve them, there is a market for cities, pest control companies, HOAs, government agencies, and academic researchers to be contributing members of the Pest Patrol community.

4 Pest Patrol Prototype Description

4.1 Prototype Architecture (Hardware/Software)

How will the prototype be structured to demonstrate key features of the CS 410 product? Describe the Prototype MFCD.

4.2 Prototype Features and Capabilities

What does the prototype demonstrate?
Why is that significant in showing how the problem is solved?
How have you demonstrated success?
How does the prototype address the CS 410 project risk mitigation?
Describe the functional goals and objectives.

4.3 Prototype Development Challenges

Describe the expected challenges to be encountered while completing the prototype – e.g., knowledge missing, capability missing, supporting technology issues.

5 Glossary

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

Community Member: A member of a community, see Community definition

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

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

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

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

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

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-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