Lab 1 – Pest Patrol Product Description

Tom Cooch

Old Dominion University

CS 411W, Fall 2022

Professor J. Brunelle

October 12, 2022

Final Version

Contents

Figures & Tables

1. Introduction

Pests have maintained a constant presence in human society throughout its entire existence and are an unavoidable fact of life. Although most interactions are benign and carry no long-term ramifications, the more serious incidents can have major negative impacts to an individual's health, property and/or finances.

Common community pests such as raccoons, mosquitos and ticks are known to be vectors of diseases. Diseases like rabies which can be carried by raccoons and the West Nile virus which can be carried by mosquitos are potentially fatal and others like Lyme disease which can be carried by ticks are chronic and debilitating. In total, they are contracted by ~300,000 people annually (CDC, How many people get Lyme disease?, 2021). These diseases alone also impose significant financial burdens as treatments cost the United States more than 1.8 billion dollars annually, with West Nile Virus costs reaching $57 million (Doucleff, 2014), rabies costs ranging up to $510 million (CDC, 2019), and Lyme Disease costs totaling up to $1.3 billion a year (Desmon, 2015). The overall health costs for all pest-borne diseases is undoubtedly higher.

Based on household data from 2020, property costs related to pest removal and damages are both wide reaching and potentially expensive. In 2020, of the 128.45 million households in the US (Department, 2022), 12.8 million had problems with roaches and/or rodents (Sellner & Wicht, 2021) while another 600,000 were infested with termites (Parkman, 2022). On the low end, the respective exterminator costs to a single household for roach and rodent removal starts at $350 (HomeAdvisor, How Much Does Fumigation Cost For Roaches?, 2021) and $354 (HomeAdvisor, 2022). Property damages add another minimum $150 to the total but can far exceed $4,500 (HomeAdvisor, 2022). For termites, households spend between $1,000 and

$3,000 just for extermination services (Termite Statistics, n.d.), and the collective annual cost exceeds $5 billion (Termite Statistics, n.d.).

Over the last several decades, these statistics have been increasing overall (Fernandez, 2021). The primary cause is due to the current warming trend of the planet. Warmer winters result in less insect die off and greater populations in the spring and summer (Guard, 2018). This has also simultaneously led to a reduction of biodiversity – available food sources – for predatory animals (Abrahms, 2021). The lack of resources combined with habitat destruction from human expansion has resulted in more contact with these forms of wildlife (Abrahms, 2021).

Many scientists predict our planet's warming pattern will continue for the long term (UCAR, n.d.). Under these conditions, communities can expect more unwanted interactions with pests. Although this expectation is likely, it is not a certainty. It is possible to change the status quo but doing so involves addressing its inherent problems.

One of the main problems is that communities have no reliable way of staying informed on local pest encounters. This information is not consolidated in a centralized platform, which means reports about pests are fragmented over multiple social media applications like Facebook and/or Nextdoor. These applications also cater to a generalized audience and have no specialized functionality for pest tracking. These two factors decrease the likelihood that all incidents are being reported. Being fully apprised on the information available involves actively monitoring multiple applications concurrently, which is not feasible.

Another problem is that a lack of real-time awareness results in pest encounters that would otherwise be preventable. The lack of geolocational data makes it difficult for individuals to pinpoint where pest encounters are occurring even if they are reported on an existing application. Even if the location is communicated, none of the applications translate the information to a geolocational tag on a map. They also are not able to alert users of potential impending danger if they happen to be close to it – especially if the report occurred within the last 10-15 minutes. Users would need to have the application open, read the information posted, and triangulate the location relative to their own. The more likely outcome is they fail to receive the necessary information in time to avoid the encounter. They enter the situation unaware of the danger and are at greater risk for bodily harm.

The lack of effective tracking in the present also extends to the past which highlights a third problem. The experience and lessons learned from previous pest issues are not retained. Many pest issues currently affecting a community have likely occurred before but were not catalogued. Any valuable information that could be leveraged by other community members is lost. Without this data, the baseline response to future occurrences fails to improve.

Finally, when planning outdoor activities or trips to other locations, it is difficult to effectively prepare for active pest problems because the information does not exist. Individuals are unable to modify their preparations according to the current threats or they fail to make alternate arrangements when it is more cost effective to do so. This puts them at greater risk of bodily harm in addition to a loss of time and money.

Pest Patrol is a software solution that aims to protect neighborhoods and communities from the pests that inhabit them by promoting localized information exchanges and transparency. Pest Patrol users will be informed of pest encounters in their area through user-generated updates of sightings and encounters, which will in turn prevent or limit future pest encounters. This result is also supported through the application’s heat-mapping feature, which highlights areas with significant user reports indicating a potential outbreak. The application will also aggregate data from previous user posts to assist users with future incidents. Lastly, the application will include up-to-date, location-specific data that will enable users to make informed preparations for outdoor activities or trips.

2. Pest Patrol Product Description

The Pest Patrol mobile and PC application relies on aspects of social media to crowdsource pest-related sightings and incidents to increase public awareness of potential threats in their area. Users can offer guidance to one another regarding pest types and solutions for managing pest problems. These user contributions build a knowledge base for future users to draw upon when faced with similar pest situations. Heat-mapping and predictive modeling are also central features of the application. These features increase awareness among users about potential outbreaks or significant pest issues within their community, which can provide insights to them and pest control companies enlisted to mitigate pest-related issues.

Pest Patrol's overarching goal is to cultivate safer communities by minimizing unexpected pest encounters. While the application may not prevent all pest-related incidents, the aim is to mitigate these potential issues and lessen the severity of problems that do occur. To carry out these goals effectively, the app will have a streamlined, intuitive interface for users to report and learn about pest incidents in their community or within an area of interest. To ensure users have actionable information, users will tether their sightings and reports to a precise location. The interface will also facilitate user-to-user communication, with these discussion threads and posts being aggregated into a user's feed.

2.1 Key Product Features and Capabilities

As a web-based application, Pest Patrol is available on all devices and operating systems. Although it is accessible without creating an account, non-registered users are limited to view-only access; all interactive functionality is disabled. The account setup process is relatively basic in terms of mandatory information. New users are only required to provide a valid email address, password, and home community. Upon verifying their email address, the interactive capabilities are unlocked.

One of the key features of Pest Patrol are the pest alerts. Pest alerts are real time notifications that trigger whenever there is a new pest incident within the user's vicinity, and they help them avoid unwanted pest encounters. These alerts are sent to the mobile device and appear as a Pest Patrol logo in the notifications section. Upon initial receipt, the user will either receive an audible cue from their mobile device or it will vibrate. The user can determine which manner they wish to receive the notification or decline receiving any notification. They can further customize alert settings based on the type of pest, and they can define a geographical radius – around their location – from which the alerts will trigger. Due to its limited utility on the PC, the alert notification is only available for mobile devices. It is the only feature not available between the two versions.

Once users have logged into Pest Patrol, the first interface that loads is the dashboard. The dashboard is the foundational piece of the application on which all features are rooted; it provides access to every feature within Pest Patrol. The dashboard display can be set to one of the three display modes: Incident Map, Discussions, and Hybrid. The Incident map displays a map of the community and reported pest incidents, Discussions display the discussion threads related to the incident reports, and Hybrid mode divides the interface to display both the incident reports and discussion threads together. As the dashboard initializes, the application will compile all new notifications – thread discussion updates, new messages, new reported incidents, etc. – for the user's convenience. These updates will appear as updates under the "Recent

Neighborhood Activity" section of the dashboard. In addition, the user will receive A.I generated tips, predictive modeling based preemptive alerts, guides, and recommended incident discussions based on their activity and historical pest activity.

The Incident Map is the main interactive feature of Pest Patrol, and it provides users a comprehensive view of all reported pest threats in their area. With just a glance of the map, users can gain complete awareness of what pests are in their community and their locations. If users are not satisfied with the default settings, the incident alerts that appear can be customized based on discussion thread content (based on key words), user id of the incident submitter, pest type, and age of the report. These customizations can be saved in the user's profile settings or applied temporarily using the ad hoc controls in the dashboard panel. Each reported incident can be clicked or tapped to open a pop-up interface that displays all related user discussion threads. The incident map can also be set to an alternative heat mapping display. This alternative view displays pest activity over a set period of time and highlights where pest activity is most prevalent; it is useful for identifying potential emerging pest problems. New reported incidents are also submitted through the incident map.

The Discussion view serves two primary purposes. It provides quick at-a-glance information related to the reported incidents on the Incident Map, and it acts as the historical repository for all discussion threads that have occurred within a community. Users with current pest problems can leverage this information to devise more effective mitigation strategies. This information is also ingested by a machine learning API with the intent of providing additional assistance/guidance. Users can also choose to follow discussions they are either actively engaged or are interested. Followed discussions are automatically made available through the dashboard interface for convenience purposes. Users will also be notified whenever one of these threads has new activity.

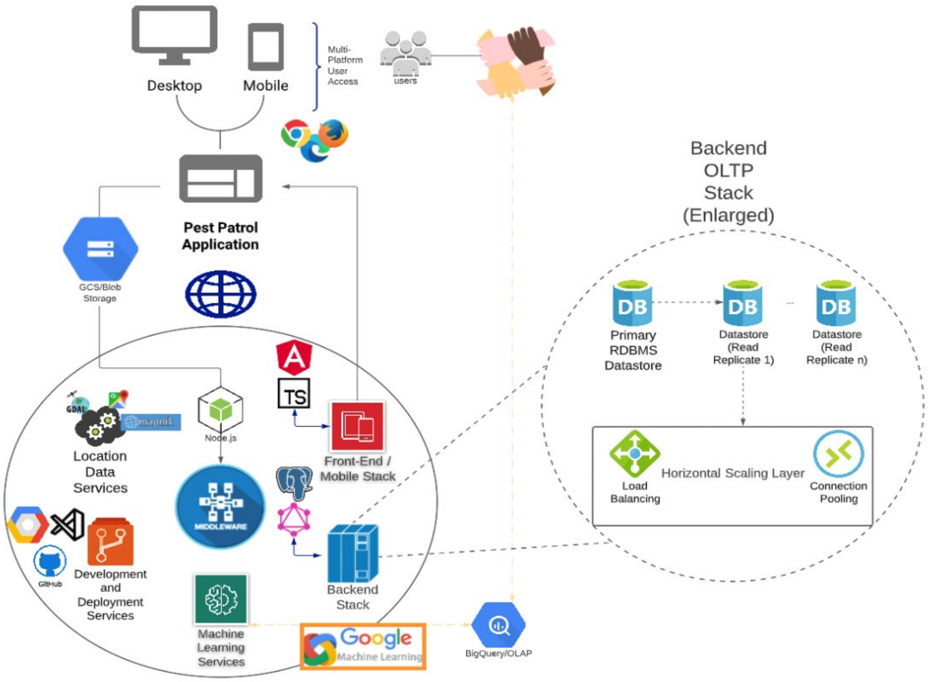
Aside from the discussion threads, other community interactive aspects of Pest Patrol are its direct messaging and community features. The direct messaging interface is a straightforward interface for users to contact one another directly. The community feature is accessible through the dashboard interface and lets users add each other as friends. Another feature dedicated to user interactivity is A.I based bot moderation. The purpose is to police inappropriate conduct as well as inappropriate application use (e.g. deliberately reported erroneous pest incidents).

2.2 Major Components (Hardware/Software)

Pest Patrol is a web application that is accessible with any modern web browser. The major functional component diagram in Figure 1 illustrates the process by which front-end input reaches back-end storage and the key components involved.

Figure 1

Pest Patrol Major Functional Component Diagram



The application consists of PC and mobile front-end interfaces for user input and back-end storage components for storage. The front-end interfaces will run primarily on JavaScript and it will use Node.js as the middleware by which all components displayed in the MFCD will communicate. The back-end data stores will be a google based cloud storage solution. Transaction text-based data will be accessed with PostgreSQL and will be available in cloud SQL instances. Geo-locational features will leverage Google's Maps and API services and GDAL. The machine learning aspects will be based on Google's TensorFlow platform.

Development of Pest Patrol will use Microsoft Visual Studio Code as the source code editor. Github will be used for the collaboration environment and implementation tracker.

3. Identification of Case Study

Pest Patrol is a community resource meant to increase the collective awareness of potential pest-related threats. As such, individual community members are a significant user group. Within this user group are hikers, campers, and outdoor enthusiasts. Businesses with outdoor work environments, cities/municipalities, and homeowners associations (HOAs) would also be members of Pest Patrol's targeted audience. Table 1 outlines these groups' motivating self-interests and examples of messages that cater to those interests.

Table 1

User Group Communications

User Group	Motivating Self-Interests	Key Message
Community Members	Personal safety, potential risks pets and children	Download Pest Patrol to keep yourself and those you love safe from potential pest-related threats including stings, bites, and diseases.
Hikers, Campers, Outdoor Enthusiasts	Proper planning for outings, safety	Know before you go! View Pest Patrol's map feature for latest pest sightings along your route. Use this info to pack accordingly.
Businesses with Outdoor Work Environments (e.g., landscapers, loggers, dog walkers)	Employee and customer or client safety, minimizing potential negative impacts to business due to pest-related incidents	Avoid project delays by preparing your employees for potential pest-related threats with updates from Pest Patrol. Avoid negative reviews and keep your clients safe by staying up to date on pest-related threats in your area with Pest Patrol.
Cities/Municipalities, Homeowners Associations	Overall community safety	Identify emerging pest-related threats and opportunities for community engagement and education with updates from Pest Patrol.

In addition to the primary user groups, Pest Patrol has a broader applicability to users interested in using the app for data-driven decision-making. Pest control companies could leverage Pest Patrol data to inform their marketing efforts. For example, a pest control company could identify hot spots where mosquitoes are highly prevalent. Based on this, the company could initiate a door hanger campaign in that specific area or a direct mailer campaign offering a unique promotion for mosquito treatments. Homeowners associations would also be invested in

Pest Patrol's data as they have a vested interest in community safety and maintaining a neighborhood's positive reputation.

Lastly, Pest Patrol data could also be used for scientific purposes. Government agencies, such as the U.S. Fish and Wildlife Service, could monitor sightings for reports of invasive species, allowing them to track and potentially address these pests. More broadly, researchers could use Pest Patrol data for population tracking studies or other research purposes.

Pest Patrol will be tested using the Hiker, Camper, and Outdoor Enthusiast user group. The team will recruit participants using Facebook Ads targeted toward outdoor enthusiasts with an interest in camping, hiking, outdoor recreation, and hunting. There are no constraints on age, demographics, or education of the targeted audience. Volunteers will be asked to take a brief survey to confirm they are interested in the outdoors and to obtain contact information. These volunteers will then participate in a virtual one-on-one user testing session to assess their thoughts about the Pest Patrol concept, determine usability of the app, and gather user feedback on the experience.

4. Pest Patrol Product Prototype Description

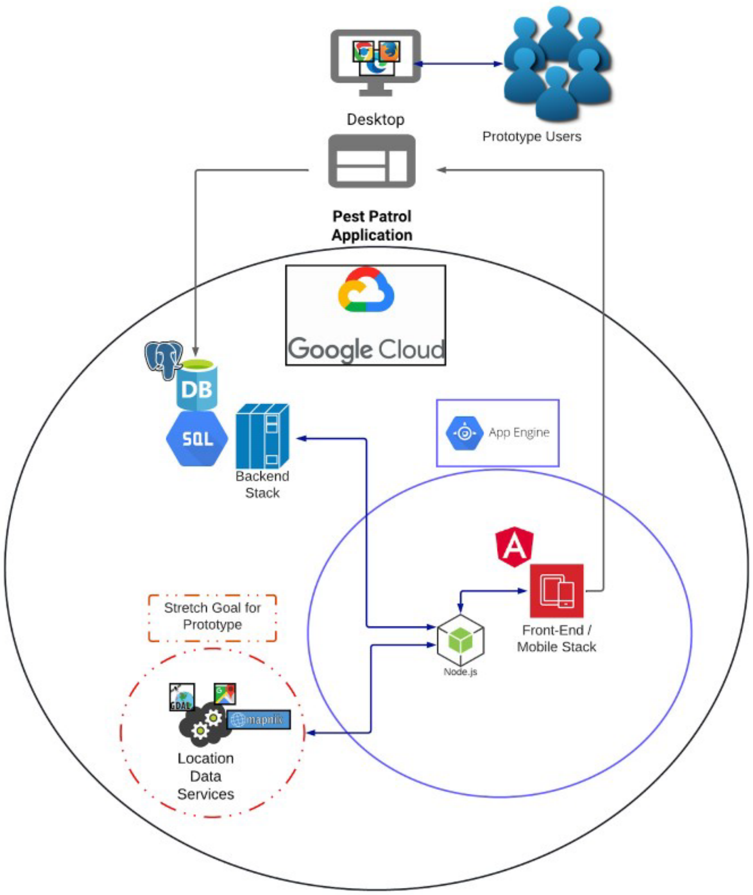
The primary objective of Pest Patrol is to eliminate or minimize the negative impacts of pests within communities. Not all features from the real-world product will be included in the prototype, but despite these differences, it will still demonstrate that the core features achieve Pest Patrol's intended purpose. Users will benefit from the aggregated knowledge of previous experiences to become better informed about pests and active pest problems in their community. They will also be able to contribute to the knowledge base themselves and collaborate with other community members about pest issues.

4.1. Prototype Architecture (Hardware/Software)

Pest Patrol is a web-based application which should ensure that it is compatible across all operating systems and devices. The only baseline requirement is the ability to run a modern internet browser. The prototype functionality will be based on the MFCD diagram outlined in Figure 2. All aspects of the application will be hosted using Google Cloud services.

Figure 2

Prototype Major Functional Components Diagram



The Pest Patrol prototype application's front-end will be implemented using the Angular framework, and it will be deployed on the Google App Engine. The former will facilitate the creation of responsive user interfaces that can effectively adjust to the wide array of display resolutions, and the latter will reduce infrastructure management and related issues. Back-end resources will include an instance of PostgreSQL as the data warehousing solution and possibly Location Data Services to support the geolocational features. The integration of Location Data services will be contingent on how quickly development of the other components progress. Communication between the front-end and back-end will rely on Node.js middleware which will also run on the Google App Engine.

4.2. Prototype Features and Capabilities

Although some capabilities were omitted or scaled down in the planned prototype's design, they will have a negligible impact on the performance of its core features; the goals and range of functionality outlined in section 2 will still be realized. A complete listing of each feature and its implementation status can be viewed in Table 2, and each function is grouped according to one of the core features (General, Incident Map, Discussion View, Pest Alerts and Community) it supports.

Table 2

Pest Patrol RWP vs. Prototype Feature List

Function	Real World	Prototype
General		
Web and mobile compatibility	Fully Functional	Partially Functional
Dashboard	Fully Functional	Fully Functional
Hybrid Mode	Fully Functional	Eliminated
Authentication and Identification	Fully Functional	Eliminated
Password Recovery	Fully Functional	Eliminated
Incident Map		
Incident Map	Fully Functional	Fully Functional
Incident Reporting	Fully Functional	Partially Functional
Ad hoc Incident Filtering	Fully Functional	Fully Functional
Heat Mapping	Fully Functional	Partially Functional
Discussion View		
Discussion Thread View	Fully Functional	Fully Functional

Expanded discussion view	Fully Functional	Fully Functional
Follow/Subscribe to discussion thread	Fully Functional	Fully Functional
Discussion thread creation	Fully Functional	Fully Functional
Reply to discussion thread	Fully Functional	Fully Functional
Provide positive/negative feedback to threads	Fully Functional	Fully Functional
Pest Alerts		
Pest Alerts	Fully Functional	Partially Functional
Alert customization	Fully Functional	Partially Functional
Community		
Search for user	Fully Functional	Fully Functional
Add friends	Fully Functional	Fully Functional
Report Users	Fully Functional	Fully Functional
User reputation system	Fully Functional	Eliminated
Automated Moderation (ML)	Fully Functional	Eliminated
Hide flagged content	Fully Functional	Fully Functional
Account suspension	Fully Functional	Fully Functional
Flag inappropriate content	Fully Functional	Fully Functional
Content removal	Fully Functional	Fully Functional
View flagged content	Fully Functional	Fully Functional
Block user	Fully Functional	Fully Functional
Content search	Fully Functional	Fully Functional
Recent Neighborhood Activity	Fully Functional	Fully Functional
Direct Messaging	Fully Functional	Fully Functional
New thread activity notification	Fully Functional	Fully Functional
New direct message activity notification	Fully Functional	Fully Functional
New incident notification	Fully Functional	Fully Functional
AI generated notifications (ML)	Fully Functional	Eliminated
Notification customization	Fully Functional	Fully Functional
Predictive Modeling (ML)	Fully Functional	Eliminated

The initial planned implementation would see the product take the form of a PC and mobile application but completing both under the time constraints would be unlikely. Instead, the prototype's development framework was switched to a web-based model, and the change in development strategy will improve the accessibility.

Although the incident reporting and heat mapping functions of the incident map feature will only be partially implemented, the process of reporting, accessing, and viewing incidents will be commensurate with the real-world product. The limitations apply to the granularity by which users can report incidents on a map. Heat mapping functionality will also be more limited due to the required level of effort for full implementation of the mapping feature.

Another feature that will only see partial implementation is the Pest Alert. While there would be other challenges to completing this development, the primary issue is with the APIs. As a web-based application, Pest Alerts transmitted to mobile devices will take the form of SMS notifications, but due to limited availability of the underlying API's, they are not a viable option. The planned workaround to the issue is simulating the SMS notifications with JavaScript alerts. This also affects the alert customization function since it also assumed use of the same API. Alert customizations are still planned for the prototype, but they will be carried out using JavaScript.

The discussion view feature in the prototype will be fully implemented as it would with the real-world product. Users will be able to access discussion threads through the incident map or through the dedicated discussion interface on the dashboard. They will be able to respond to existing threads and create new threads. For easier viewing, discussion threads can be expanded to occupy the entire dynamic viewing portion of the dashboard. It will also be possible for each user to rate a thread as positive or negative which will affect its overall quality rating.

Barring the AI/Machine learning based functions, the prototype's community feature will be fully functional. Users will have access to functions available in most social media applications like adding other users as friends, blocking users, and direct messaging. They will be able to report users for inappropriate behavior and it will be possible to suspend accounts based on conduct. Other functions like community notifications and thread content searches will also be available. The planned AI element was meant as an enhancement to the existing features.

Tasks like user interaction and content moderation and content searches contain repetitive processes that an AI could automate.

Eliminated features include the hybrid display mode and user authentication and AI/machine learning processes. While these elements would enhance the overall user experience, they were extraneous aspects of their associated core feature, and the necessary developmental effort would likely outweigh the benefits provided.

4.3. Prototype Development Challenges

From a technical standpoint, the main challenge is the limited collective knowledge and experience with the development tools. In addition to the learning curve associated with the prototype's underlying programming languages, there is an additional learning curve associated with the development tools themselves. It seems likely that these factors would impede forward progress in the early development phases. These potential delays also contribute to already existing concerns about the scope of the project and the feasibility of completing it within the given timeframe.

There are also additional challenges with coordinating the project with a geographically decentralized team while also juggling the development of a front-end, middleware, and back-end simultaneously. Coordinating schedules between team members across multiple time zones is already a considerable challenge, and it is difficult to envision how these three components will be integrated smoothly.

5. Glossary

Administrator: Individuals responsible for keeping an application running. For Pest Patrol these responsibilities include the moderation of user interactions and posted content.

Angular: a typescript based open-source web development framework

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

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

CRUD: Create, Read, Update, and Delete. They are the four basic operations of persistent data storage.

Density: The number of incidents linked to a certain pest in a given area

Discussion Thread: running commentary of messages between members within a community

Docker: an application that uses operating system level virtualization to deliver software in packages called containers

Feedback: a positive or negative rating that can be applied to any user-created discussion thread or response

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

Git: a distributed version control system for software development

GitHub: an internet hosting service for software development and version control using Git

Google App Engine: a cloud computing platform for the development and hosting of web applications in Google-managed data centers.

Google Location Services: A service from Google that aims to provide a more accurate device location and generally improve location accuracy.

Heat Map: a data visualization technique that shows the magnitude of a phenomenon as a color in two dimensions

Hiker/Camper: Any individuals that engage in prolonged leisure activities outdoor

Homeowner’s Association (HOA): A membership organization formed by a real estate developer to own and maintain common green areas, streets, and sidewalks and to enforce covenants to preserve the appearance of the development. It is operated for the benefit of all the residents of the community.

Incident Map: a graphical map that displays the locations of all reported pest incidents for an area

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

Infestation: A specially designated incident involving a high concentration of pests in a given area

Instance: a single copy of the software running on a single physical or virtual server

Interface: frontend graphical displays of the Pest Patrol application that users interact with

Major Functional Component Diagram: a high-level visualization of the main system resources and their dependencies with one another

Node.js: an open-source, cross-platform, back-end JavaScript runtime environment that runs on a JavaScript Engine and executes JavaScript code outside a web browser

Pest Control Company: any business entity that specializes in the regulation or management of pest species

Pest Hotspot: A number of reported incidents of a specific pest that exceed a certain threshold determined by the user

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

PostgreSQL: an open-source relational database management system emphasizing extensibility and SQL compliance

Predictive Modeling: the use of statistics to predict outcomes

SMS Messaging: Short Message Service. Is a text messaging service that uses standard communication protocols to enable the exchange of short text messages

TypeScript: Open source programming language that is a syntactical superset of JavaScript.

User: Any individual interacting with the Pest Patrol application. This includes: community members, hikers/campers, pest control companies, homeowner’s associations and administrators

Virtual Machine: a compute resource that uses software instead of a physical computer to run programs and deploy apps

VSCode: a source-code editor made by Microsoft that includes features for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

6. References

Abrahms, B. (2021, July 30). *Human-wildlife conflict under climate change*. Retrieved from Science: https://www.science.org/doi/10.1126/science.abj4216

CDC. (2019, June 11). *Cost of Rabies Prevention*. Retrieved from Centers for Disease Control and Prevention: https://www.cdc.gov/rabies/location/usa/cost.html

CDC. (2021, January 13). *How many people get Lyme disease?* Retrieved from Centers for Disease Control and Prevention: https://www.cdc.gov/lyme/stats/humancases.html

Department, S. R. (2022, February 22). *Number of households in the U.S. from 1960 to 2021*. Retrieved from Statista: https://www.statista.com/statistics/183635/number-of-households-in-the-us/

Desmon, S. (2015, February 5). *Lyme Disease Costs Up to $1.3 Billion Per Year to Treat, Study Finds*. Retrieved from John Hopkins Bloomberg School of Public Health: https://publichealth.jhu.edu/2015/lyme-disease-costs-more-than-one-billion-dollars-per-year-to-treat-study-finds

Doucleff, M. (2014, February 12). *The High Cost of Treating People Hospitalized With West Nile Virus*. Retrieved from NPR: https://www.npr.org/sections/health-shots/2014/02/11/275262857/the-high-cost-of-treating-people-hospitalized-with-west-nile-virus

Fernandez, L. (2021, July 7). *Market value of the United States pest control industry from 2010 to 2021*. Retrieved from Statista: https://www.statista.com/statistics/1134191/us-pest-control-market-size/

Guard, V. (2018, January 15). *How Warmer Winters Affect Pests*. Retrieved from Varment Guard Wildlife Services: https://varmentguard.com/blog/warmer-winters-affect-pests

HomeAdvisor. (2021, June 11). *How Much Does Fumigation Cost For Roaches?* Retrieved from HomeAdvisor: https://www.homeadvisor.com/cost/environmental-safety/roach-fumigation/

HomeAdvisor. (2022, January 10). *How Much Does Rodent Control Cost?* Retrieved from HomeAdvisor: https://www.homeadvisor.com/cost/environmental-safety/rodent-removal/

Parkman, K. (2022, February 17). *Pest control statistics and trends*. Retrieved from ConsumerAffairs: https://www.consumeraffairs.com/homeowners/pest-control-statistics.html

Report, S. (2019, July 1). *Vector-Borne Diseases on the Rise*. Retrieved from Meigs Independent Press: https://meigsindypress.com/2019/07/01/vector-borne-diseases-on-the-rise/

Sellner, M., & Wicht, J. (2021, April 21). *Residents of 14 Million Housing Units Reported Seeing Roaches, 14.8 Million Saw Rodents in Last 12 Months*. Retrieved from United States Census Bureau: https://www.census.gov/library/stories/2021/04/how-many-american-homes-have-pests.html

Termite Statistics. (n.d.). Retrieved from Orkin: https://www.orkin.com/pests/termites/termite-statistics

UCAR. (n.d.). *Predictions of Future Global Climate*. Retrieved from Center for Science Education: https://scied.ucar.edu/learning-zone/climate-change-impacts/predictions-future-global-climate