CIS\*3750 Assignment 1

Veto Petition

Ebad, Babar

James, Burns

Mitchell, Coakley

Ralph Arvin, De Castro

Josh, Guenther

Table of Contents

|  |  |
| --- | --- |
| Client Details | 2 |
| Project Goals & Users | 3 |
| Definitions | 4-5 |
| References | 6 |
| Individual Contributions | 7-11 |
|  |  |
|  |  |

**Client Details**

Wellington-Dufferin-Guelph Public Health (WDGPH) is a public health unit in Ontario, providing various health services in the city of Guelph, and Dufferin and Wellington counties. The proliferation of ticks in Ontario has raised concerns among the citizens and public health officials as ticks, more particularly, the black-legged tick, could transmit a bacteria from the genus *Borrelia* in humans. Being bitten by such tick could start the onset of a serious illness, called Lyme disease. Lyme disease could negatively affect the overall health and quality of life of an infected human. The WDGPH is doing their best to educate and spread awareness about lyme disease and black-legged ticks.

Usually, people who want to report about ticks or being bitten by ticks will go to their respective public health unit to assess the situation. This includes identifying the species of the tick and the location of encountering such tick. The issue with this is that it is difficult for a person to bring a tick in alcohol and sometimes, the tick could still be alive. This process is inconvenient for both parties. Because of this, one of the WDGPH goals is to find a more convenient and faster way to identify ticks.

WDGP operates four offices in the region. They serve roughly 272,000 people (statsCan, 2019), the majority of whom are located in rural areas.WDGP is 1 of 35 public health units in Ontario(Lyme Disease Map, 2019). However, they are the only ones that deal with tick identification(Public Health WDP, 2019). Aside from tick identification, WDGH also runs immunization tests, sexual, travel, and dental health clinics in the area. They monitor environmental hazards and respond to public health emergencies. WDGPH is also responsible for monitoring food vendors, pools, spas, salons, tattoo parlors, and infection control practices of healthcare professionals(Public Health WDP, 2019).

**Project Goals & Users**

The goal of this project is to design a mobile application that identifies ticks that are known carriers of lyme disease. The application will comprise the following core functionalities: photographing a tick, submitting said tick, including relevant location and other information, and receiving a response that indicates whether the tick is a known carrier of lyme disease.

Relevant users include the student team who will design the application, the WDGPH staff who are the client and will be the administrators of the application, and the general public who will be the users of the application. The majority of the relevant demographic are between the ages of 15-64 with a slightly higher number of females than males (Statistics Canada, 2016).

Dufferin-Wellington-Guelph region, where the target audience is located, has a population of 272,000. The City of Guelph comprises 120,000 of this number, followed by the County of Wellington (without Guelph) at 95,000, and the county of Dufferin with 57,000 (WDGPH, n.d.).

People living near the border of Halton and Hamilton are most likely to use this app, as these regions are most at risk of encountering said ticks. However, it will not be uncommon for people everywhere in the region to use the app to report ticks (Public Health Canada, 2019).

This team project will be organized through the use of Google applications such as Google Docs and Google Sheets for writing and sharing documentation. Communication is being done through Discord, which allows us to communicate in multiple specific text channels and voice channels. This keeps each topic of our project separate from the others, for ease of finding relevant information during the development process.

The application will ideally be written with the Flutter framework (v1.9), a cross platform framework by Google that supports our supported versions of iOS and Android (Google, n.d.). It is written in the Dart language, and is cross-compiled to the target platforms, since Dart does not run natively on Android or iOS. This means that we can have a single codebase for both applications, rather than two codebases. On Wellington-Dufferin-Guelph Public Health’s end, we are designing around the assumption that a relational database will be used to store the data, such as PostgreSQL. This assumption means our data will most likely be provided in a form with the least amount of work needed for it to fit in a relational database. However, since the storage of the data on WDGPH’s end is not our responsibility, any data store would work, within reason.

**Definitions**

**Acceptable photo:** synonymous with suitable photo.

**AODA:** Accessibility for Ontarians with Disabilities Act, a statute designed to improve accessibility standards for those with physical and mental disabilities.

**API:** application programming interface, typically used to communicate with other software.

**Apple’s Human Interface Guidelines:** design principles and themes described by Apple for iOS. See <https://developer.apple.com/design/human-interface-guidelines/ios/overview/themes/> for more information.

**Application:** the mobile application for the supported versions of iOS and Android. Synonymous with system.

**Database:** any store of data utilized by WDGPH, from relational databases to raw data.

**Full session:** the amount of time (5 min) that the user approximately needs to begin and complete a submission.

**Google Maps API:** the protocol to use Google Maps, which can provide location-based services and functionality to the system. See <https://developers.google.com/maps/documentation/> for more information.

**Google’s Material Design:** a visual language designed by Google to ensure a familiar and intuitive experience across Android apps. See <https://material.io/design/> for more information.

**GPS spoofing:** the act of faking your GPS location, often used to circumvent geo-fencing restrictions.

**eTick:** a tick monitoring platform run by faculty at Bishop’s University Biology Department. See <https://etick.ca> for more information.

**Jurisdiction:** the area that WDGPH is required to cover.

**OpenStreetMaps API:** the protocol to use OpenStreetMaps, which can provide location-based services and functionality to the system. See <https://wiki.openstreetmap.org/wiki/API_v0.6> for more information.

**Privacy Policy:** the system will only collect information pertinent to the use of the application, and nothing else. This means information such as medical information, banking information, or other sensitive information are not to be collected. Examples of information that are pertinent would be user location, as well as contact information.

**Questionnaire:** the form required by WDGPH to be provided with all submissions.

**Response:** responses from WDGPH can be in the forms of (1) email, (2) phone call, and (3) in-application. Our system only needs to handle the in-application response, which could be a simple notification and message.

**SQL Injection:** Inserting malicious SQL commands in input boxes designed to specifically remove or edit table contents in a database

**Submission:** the combination of tick image, location data, and questionnaire provided by WDGPH.

**Suitable photo:** a suitable photo is a photo that is able to effectively be analyzed by WDGPH staff. A suitable photo follows the following criteria:

* The tick should be central to the image (ideally the very center, but off-center is acceptable). The tick should NOT be along the edge of the image.
* The tick should be clearly visible to the naked eye. The tick should NOT blend into the background. For example, if the tick is black, do NOT take an image with the tick in-front of a black background.
* Appropriate lighting should be present. The image should not be pitch black, but also not blindingly bright. To ensure a sufficient brightness, try to make the setting brighter than you would need to see the tick in-person. Images are often darker than their setting.
* The image should have sufficient quality. To achieve sufficient quality, take the image at least a foot away from the tick, up-to at most a few feet away. Minimizing the use of zoom generally results in a less-distorted photo.
* A sense of scale is ideal. For an appropriate object to use, consider a standardized, common object like a Canadian quarter or other unit of currency.

**System:** the mobile application for the supported versions of iOS and Android. Synonymous with application.

**Tutorial:** a set of instructions that provides basic information and functionalities of the application to familiarize the users.

**Two-factor authentication:** a combination of any two sets of credentials, (1) an email address, (2) a username/password combination, (3) a mobile phone number.

**WDGPH:** Wellington-Dufferin-Guelph Public Health, our client.

**References**

Google. (n.d). Flutter Documentation. Retrieved September 27, 2019, from <https://flutter.dev/>

Public Health Canada. (2019). Ontario Lyme Disease Estimated Risk Areas Map: 2019. Retrieved

September 26, 2019, from

<https://www.publichealthontario.ca/-/media/documents/lyme-disease-risk-area-map-2019>

Statistics Canada. (2019). Census Profile, 2016 Census - Wellington-Dufferin-Guelph Health Unit.

Retrieved September 26, 2019, from

[https://www12.statcan.gc.ca/census-recensement/2016/dp-pd/prof/details/page.cfm?Lang=](https://www12.statcan.gc.ca/census-recensement/2016/dp-pd/prof/details/page.cfm?Lang=E&Geo1=HR&Code1=3566&Geo2=PR&Code2=35&SearchText=Wellington-Dufferin-Guelph%20Health%20Unit&SearchType=Begins&SearchPR=01&B1=All&GeoLevel=PR&GeoCode=3566&TABID=1&type=0)

[E&Geo1=HR&Code1=3566&Geo2=PR&Code2=35&SearchText=Wellington-Dufferin-G](https://www12.statcan.gc.ca/census-recensement/2016/dp-pd/prof/details/page.cfm?Lang=E&Geo1=HR&Code1=3566&Geo2=PR&Code2=35&SearchText=Wellington-Dufferin-Guelph%20Health%20Unit&SearchType=Begins&SearchPR=01&B1=All&GeoLevel=PR&GeoCode=3566&TABID=1&type=0)

[uelph%20Health%20Unit&SearchType=Begins&SearchPR=01&B1=All&GeoLevel=PR](https://www12.statcan.gc.ca/census-recensement/2016/dp-pd/prof/details/page.cfm?Lang=E&Geo1=HR&Code1=3566&Geo2=PR&Code2=35&SearchText=Wellington-Dufferin-Guelph%20Health%20Unit&SearchType=Begins&SearchPR=01&B1=All&GeoLevel=PR&GeoCode=3566&TABID=1&type=0)

[&GeoCode=3566&TABID=1&type=0](https://www12.statcan.gc.ca/census-recensement/2016/dp-pd/prof/details/page.cfm?Lang=E&Geo1=HR&Code1=3566&Geo2=PR&Code2=35&SearchText=Wellington-Dufferin-Guelph%20Health%20Unit&SearchType=Begins&SearchPR=01&B1=All&GeoLevel=PR&GeoCode=3566&TABID=1&type=0)

Wellington-Dufferin-Guelph Public Health. (n.d.). Region of Wellington, Dufferin and Guelph.

Retrieved September 26, 2019, from

<https://www.wdgpublichealth.ca/your-community/emergency-preparedness/emergency-response-plan/region-wellington-dufferin-and-guelph>

Individual Contribution: Ebad Babar

**Problems and Solutions:**

Exact Instruction Challenge say the kids struggle to explain how to make a pb and j sandwich to their dad. One of the first issues the kids ran into was assuming that their dad already knew how to make the sandwich and thus gave him a rough guideline of the ingredients and the steps required. Even though their dad asked for very specific instructions. For instance, the kids told their dad to put the peanut butter on the bread with no with instructions on how much and how to spread it. Another problem the kids was their instructions used undefined terminology. They used words like “a bit”, “top” and “all over” which aren’t measurable and can mean different things to different people. They ran into the problem where their dad “the user” had to guess what they meant. Lastly, the kids instructions lacked details causing them leave critical information out. An example of this was “take the knife with peanut butter and spread it on bread” leaving out the part about scooping the peanut butter out of the jar.

**Similarities to the development process:**

These problems are similar to the ones encountered while developing a requirement document as attention to details, defined terminology, and not making assumptions can help both the client and the developer thoroughly understand the problem and the proposed solution.

**Opinion**

I disagree with the statement that “requirements are a set of instructions to build something”. Requirements are the problems or what needs to be done for the project; they don’t tell us how to solve the problem.

Individual Contribution: James Burns

**General Problems and Suggested Solutions:**

1. The children were not specific in their instructions, instead providing generalized instructions that heavily implied how to do said tasks. Depending on who is supposed to follow these instructions, they may or may not be able to correctly follow the real, implied instructions. In the case of the father, he only sometimes followed the implications, but often times followed the instructions literally. If the children were more specific in their instructions, perhaps giving smaller, but more instructions, it would be less likely that someone (their father) could misinterpret their intent.
2. The children took for granted that instructions would be followed the same way each time. Following the children’s instructions, the father would sometimes follow an instruction one way, and another time follow the same instruction a different way. Since the children never gave details on how exactly to accomplish certain tasks, the father had liberty to accomplish the task however he saw fit, provided the result was “correct”. If the children included more information on how to accomplish each step (which is ironic considering this is an instruction set itself), the father would be forced into doing the steps the same way each time.
3. The father actively attempted to find holes in their instructions, and sought to exploit these holes wherever possible. He pretended not to have as much background information as he would usually have, and went to some significant lengths to disobey their intent but follow their instructions. Honestly, no instruction set can be foolproof from malicious actors, but a helpful start could be to include what not to do in some critical steps, particularly steps that are easier to misinterpret.

**Relation to a Requirements Document:**

It should be quite clear that the problem of specificity the children faced is exactly the same problem one will face when developing a requirements document. People are not mind readers and will not always be able to understand an author’s intent. If you’re not specific in your requirements, you are leaving a significant portion of the implementation reader-defined, which could result in a completely different work than what was intended. This was clearly shown in the video, when the children expected a sandwich but were more often met with a mess.

**Requirements vs Instructions:**

A good set of requirements can be akin to a set of instructions, if the detail is great enough. However, they really are different things. Instructions care not only about the end result, but about the process that takes you there. Requirements on their own more typically follow the philosophy that “the end justifies the means”. As long as your end result checks all the requirement boxes, then it fits the requirements. If it fits, is it not the “correct” product?

Individual Contribution: Mitchell Coakley

**Problems and Solutions:**

1. Instructions were executed without anything being implied. An example of this is when the instruction “rub the jelly all over the piece of bread” resulted in the jelly being rubbed all over the bread, and not on one of its sides.

To overcome this, the instructions should not rely on any implications, and assume the user does not know what they are doing.

1. Instructions were written generally but performed specifically. “Scoop a bit of peanut butter out…” was a good instruction, but relies on the user knowing what “a bit” of peanut butter is.

If instructions were more specific and did not rely on non-definable terms, these types of problems would not happen.

1. The person executing the instructions tried his hardest to find and extrapolate flaws in the instruction set.

Whatever the kids wrote, the dad would probably have found some tiny misnomer and capitalized on it. This one is an unavoidable problem (unless of course the kids made someone else follow their instructions).

**How are these problems related to the process of developing a requirements document?**

These problems can all show up in developing a requirements document. Even if the person(s) using the requirement set are sensible people, everyone has subtle differences in the way they write and interpret data. This can result in similar problems as described above - something you can easily imply may not be seen the same way by everyone else.

**Requirements are a set of instructions to build something?**

Requirements are NOT a set of instructions to build something. Requirements are a set of criteria that can be fulfilled. Requirements describe the “What”, “Where”, and “Why” of a project, but never the “How”.

Individual Contribution: Ralph Arvin De Castro

One of the general problems that the kids encountered was that some of the steps on their instructions lack definition. For example, the kids used the word “it” to refer to an object that they have mentioned before but their father thought the word “it” in a different way. To address this, it is important to define and replace the pronoun words such as “it” with its corresponding noun to avoid confusion. Secondly, some parts of their instructions were not measurable. For example, the girl told his father to scoop a bit of peanut butter. Using phrases like “a bit of” can mean different things to each person. In this case, it is better to specify a measurable amount such as “one tablespoon of peanut butter” to get an exact amount. Lastly, their instructions were not specific enough for their audience. In this video, their father wanted very exact instructions and pretended to be a person who has never seen or made any peanut butter and jelly sandwich. In this situation, their statements has to be very specific and exact enough for their father. Otherwise, their father would follow steps differently and create a different outcome.

These problems are very related to the process of developing a requirements document since writing requirements require a level of detail in their sentences. Requirements refer to what things are needed to do in building software. It is an important duty of the team to write requirements that are exact and not vague to avoid misunderstandings while building the software.

I disagree with the following statement since requirements and instructions have different meanings. As mentioned before, requirements are the “whats” or the things that are needed to be done to build something. Requirements do not describe how to deal with a specific problem and therefore, satisfying a requirement can be done in different ways. On the other hand, instructions tell the user how something is done. The users will have the same steps of tackling a problem and also be expected to have the same outcome.

Individual Contribution: Josh Guenther

The children ran into a number of problems in their feeble attempts to provide clear enough instructions to create a peanut butter and jelly sandwich that their scrutinizing father would not be able to mess up. One of the major instructions that they consistently messed up was forgetting to tell the father which orientation to put the knife in. Also, they did not define what a side of a piece of bread was, leaving it open to the father’s interpretation. He took this in the worst possible interpretation and used the most unlikely and unforeseen option of using the crusts. They also only solved the major issue with each iteration and did not take time to extend their changes to the rest of their specifications.

Much like these children experienced, it can be difficult to make specific requirements that are still understandable by users. Unlike these kids, however, we are not allowed to lose our cool and appear frustrated. Requirements are a list of things that must be included in an application, much like what must be included in creating a peanut butter and jam sandwich.

Requirements are a set of instructions is a bit of a confusing statement, but at its core, I agree with it. While it may not be worded in the same style, they definitely share the same content. Both a requirements document and a set of instructions give a list of what needs to be done to finish an application and need to be very specific to avoid being misinterpreted by the client. Instructions are usually worded as actions, while sets of requirements are worded as statements. Instructions are not typically used to give a list of things that need to be included, but if you are willing to stretch the definition of instructions to include non-actions, I would definitely consider requirements as a set of instructions to build something.