# Welcome to the mSpray Team!

## Table of Contents

# What is mSpray?

**Welcome to the mSpray team!** mSpray is an interdisciplinary group of UC Berkeley researchers and students who are dedicated to the tracking of malaria and anti-malarial insecticides in the Limpopo region of South Africa. This project is headed by **Professor Brenda Eskenazi** of the School of Public Health, supported by Lesliam Quiros and Edmund Seto, who have moved to other universities now. These three have been the backbone of the research, and have already published a paper with the first iteration of this project, released in the spring of 2012. The paper was recently accepted into one of the largest Environmental Science magazines in the world. Neighboring regions in the nation will take on the project pending the success of this next iteration of the project, and neighboring countries like Mozambique and Swaziland have begun to show interest in adopting the project for themselves.

**So why is this such a big deal? Because malaria is a big deal. And so is DDT.** In 2012 alone, malaria-carrying mosquitos contributed to 207 million malaria victims, of which 627,000 died. In Limpopo, a particularly ravaged part of the world, the government has taken the hard approach of spraying homes with the toxic chemical DDT in order to decrease deaths. This done by sending teams of "spray workers" to enter villages armed with DDT to hose down one house at a time. This works – and repels (and kills) mosquitos at unprecedented levels for an extremely long duration of time. Unfortunately, this reprieve from the mosquito assault also comes at a price, because DDT is known to cause birth defects, and can be lethal to humans that are exposed to it at the same concentrations in which Limpopo locals are living now. To try to alleviate this issue, the South African government has recently started testing a new chemical named K-Othrine, in the family of chemicals known as pyrethroids, as an alternative to DDT. However, it is unknown whether this chemical is even an improvement from DDT. To make matters worse, the government doesn't have a simple, reliable system of measuring how often a home was sprayed, or what chemical was used. As a result, homes may be sprayed too often, or sprayed with multiple chemicals. And because some homes may be accidentally over-sprayed, some villages are accidentally left out. This fragile infrastructure also makes it hard to tell how long workers have been laboring, and makes it hard to properly pay these workers who often come from impoverished backgrounds.

So how does mSpray even begin to tackle such a massive problem? Believe it or not, it does this with a free Android app! The main product of the research is an Android application developed by previous software developers. This app is used by spray team "admins", who walk through villages with the spray team, recording how many rooms of a building are sprayed with insecticide. The app then records its GPS location, which sprayers were involved in spraying each building, and how much insecticide was used in a survey. The survey data is then immediately and automatically uploaded to Google Docs, where researchers can view it. From there, the researchers are able to, with statistical tools, decide which villages need to be sprayed next, and prevent over-spraying of homes. This smarter system for insecticide sprayings will dramatically improve the fight against malaria. This would also prevent villages from being neglected. With the logged data, researchers will also be able to track the health indices of sprayed villages, in order to determine whether K-Othrine will be a safer alternative to DDT. Finally, with the uploaded data, the government will be able to determine how much to pay each spray worker.

So once again, welcome onboard! We're excited to have you with us, and for this opportunity for us to put our minds together to solve some real world problems. So enough with the talking – let's get started!

# Christian Ground Rules

This is the section where I, your project manager, Daniel Wu, get to talk to you a little bit more personally and seriously. Ultimately, my hope for you guys is that you can use research as an opportunity to stay here. And with the opportunity to stay, I hope you'll be able to enjoy fellowship with each other and with your leaders and get equipped as ministers for the gospel the same way that I was when I got to intern at Aspera after my junior year of college. On top of that, I hope that you guys will experience the blessing of working with peers early.

With these things in mind, I want to introduce a few ground rules. Obviously, you can't be fired or punished for not doing them, but as your older brother, I want to strongly encourage you to adhere to them:

- **Never work without doing your DTs first.** As Christian ministers, nothing is more important than our daily devotions. I would rather you worked less during the day and did your devotions rather than having you skip DTs. DTs have been the key times that have helped me to stay anchored in faithfully living for God in the midst of the stress of work. Also, do your DTs so that you can do this work prayerfully. Let God's Word push you to work in ministry at least as diligently as you work in this summer job (**Let it never be said that you worked harder on this job than you did on your relationship with God, that you spent more time saving peoples' physical lives than you did being concerned for and joining in the work of saving their spiritual lives**).
    - **DT together if possible**. As much as possible. Your relationships with each other are primarily that of brothers in Christ. Affirm that commitment to each other–share with each other and encourage each other to take steps of faith, to fight sin in each other's lives, and to be faithful with your various opportunities to be Christ's hands and feet in our world.
- **If you get an offer at another CS job, take it.** This research assistant job can't replace an industry job. As an older brother, I've been so blessed with the ability to use my work experience to speak truth to career-oriented frosh, to help them see that life can't just be about job security or an advanced career. On top of that, getting a job showed my parents that I am able to take care of myself, meaning that my commitments to Christ and His ministry are not an irresponsible escape from the difficult world that we live in.
- **Never overwork yourself**. Follow normal working hours. Try not to work later than 5PM, or earlier than 9AM without the project manager's (my) permission. I'd rather you use the other hours to fellowship with each other and the underclassmen, and equip yourselves as ministers.

**Ask me for any other questions**. I am not your leader, but I am your older brother and am willing to answer any questions that you may have about work, etc. Feel free to shoot me an e-mail or GChat message at ldanielcwu@gmail.com

# How do I Get Started?

Unfortunately, no introduction manual is free of the portion detailing how to set up your computer and software to prepare for software development. We'll try to keep this as short and painless as possible, though – we promise!

If you are going to be developing on a Mac, you can feel free to ignore this paragraph, but if you're coding on an Ubuntu or Windows, you should look into the following links. (**If you're just a Windows user, you need to install Ubuntu alongside your Windows OS)**, and also look at the following links (Ubuntu installation instructions are included, so don't worry – we got you! :D):

- Installing and Configuring Ubuntu
- Setting up Compiz

For the security of our software, we're going to be using the version control system Git. **Every bit of code that you write must be contained in our private repository and not go public**. For information on how to use Git properly, please read this document:

- Intro to Git

And that's it! There may be more technology that you'll need to set up as we go (Ruby on Rails, bash scripts, etc.), but that will be dependent on your project, and you may have to learn how to do it yourself, and teach the rest of us!

# Upcoming Projects

As a massive undertaking, mSpray is going to be broken apart into multiple projects. You may be assigned to be part of multiple projects at once, or you may be called to own one of these projects. Regardless, we hope that there will be a lot of good collaboration between team members, and that everybody can learn from these projects. As a bit of a sampler, though, here are some of the projects that still need to be carried out:

---

**Project 1: Data Monitoring Website**
**Objective:** Create a website for researchers and government officials to see statistical data resulting from data collected from the mSpray application
**Skills Involved:** Ruby on Rails, HTML, CSS, Javascript, Google Maps API

**Project 2: Secure Web Data Storage**
**Objective:** Encrypt data going into Google Drive so that it's unreadable by Google itself, and can only decipherable with the proper RSA key. Accessing Drive itself should not produce legible information.
**Skills Involved:** Cryptography, Java, Google Drive API

**Project 3: Improved Mobile Interface**
**Objective:** Fix minor bugs in mobile app itself, dampen colors, make UI conform to Flat UI styles.
**Skills Involved:** Android, XML

**Project 4: Researcher-Side Software (Data Aggregator)**
**Objective:** Create client side software for downloading information from Google Drive in order to view encrypted data locally as a CSV file.
**Skills Involved:** Scripting, Google Drive API

**Project 5: Government-side Software (Paperwork Generator)**
**Objective:** Create client side software to auto-generate payroll information on official South African paperwork forms locally.
**Skills Involved:** Scripting, Google Drive API, Cryptography, Adobe Acrobat Forms, iText API

---

There will be many more projects to come after these 5, but these should be a good sampler for what needs to come next. Ultimately, the next few milestones will involve a bit of scripting, and a lot of Rails and Android.

# Rules and Guidelines

There are also some rules and guidelines that we would like to share with you just to make this research experience memorable and smooth for you. Here they are:

- Leave communication with contacts in South Africa to Brenda Eskenazi. As our main contact, and the one who knows the culture of the people in South Africa best, she is the most qualified liaison for the South African people.
- Be clear communicators with your project manager and report whenever you pass milestones or have issues.
- Follow good Git practices – that will save you the grief of commit conflicts.
    - NEVER send your code to somebody else to commit. That leads to irresponsible code and makes communication about old code impossible.
- Keep everybody else updated!
    - If you finish a project, email everybody so we can celebrate together.
    - If you learn some information that you think is helpful for others to know as well for finishing their projects, send it out in an email to everybody else.
- **Have Fun!** Be physically present when working together with other team members. Eat meals together, and feel free to take breaks together.

# HOLY CRAP THIS PROJECT IS SO STINKING HARD AND I DON'T KNOW HOW TO GO ON I HATE YOU FOR ASSIGNING THIS TO ME

While working on projects, especially software projects, you're going to come across obstacles that you don't know how to overcome. ***That is totally ok.*** Sometimes, this is because the issues themselves are literally impossible, or not worth spending the time to fix.

Whenever you find yourself racking your brain and unable to get further, try to implement the following rule: ***If you can't get any leads, and can't even attempt a solution for more than an hour, immediately email or ping your project manager***, and see what he or she thinks about the situation.

If you don't know how to approach the whole project, but think you can at least tackle one small part of it, try to do that first. Sometimes, ideas will come from other team members, Google, Stack Overflow, etc., so be creative and always be willing to try new solutions.