Chapter 1, Deliverable 1

Beach Boys
- **Patrick Amons**
- **Grant Jackson**
- **Hollande Powell**
- **Montrel Nesbitt**

Candidate 1: Drone 4 Dengue

Overview
Drone 4 Dengue is a software project that uses drone images to detect dengue mosquito breeding sites. These particular mosquitoes can transmit a disease called Dengue Fever, this disease is primarily transmitted in subtropical areas of the world.

Running and compiling
Unfortunately this project has not been updated in many years and we kept running into compatibility issues when trying to compile and run it. Ultimately due to these issues we decided not to go with this project.

Conclusion
Ultimately due to the issues listed above we decided not to go with this project.

Candidate 2: Epidemiological Modeler (STEM)

Overview
The Epidemiological Modeler (STEM) is a software tool developed to help scientists and public health officials create models and run simulations of emerging infectious diseases. These simulations can help understand and prevent diseases before they could possibly become a worldwide pandemic. We found this project interesting because we thought it related to what's going on in the world today.

Running and compiling
This project requires the Eclipse IDE in order to run along with several other Eclipse plugins. Although this project does have a useful setup page, it is lacking some information and made compiling and running this project a challenge. The first challenge we encountered was that this project requires quite a lot of plugins in order to run. The issue was standard Eclipse does not ship with the plug-in development environment, so when we tried to install the STEM plugins the "Plug-in Development" option under preferences was not there. After some research we found out we had to install the plug-in development environment from the Eclipse Marketplace. After quite a bit of time the project finished getting the required plug-ins and the number of errors

went from a few thousand down to four. In order to resolve these four errors we used Eclipse's quick fix option to install one additional library. After installing this library, recompiling the code, and completing the instructions on the setup page the program ran without any issues.

Conclusion
Although we did get this project to run without any errors, in order to run a simulation we would have to know quite a lot about pathology and epidemiology. Something none of us on this team has any knowledge of. Due to this steep learning curve we decided to not go with this project.

Candidate 3: Tanaguru Contrast Finder

Overview
Tanaguru Contrast Finder assesses the background color contrast of a website in order to make websites more accessible to people with color blindness. Tanaguru also develops an open source website assessment tool that uses apps to run tests on a website, the Contrast Finder is one of these apps that Tanaguru develops themselves.

Running and compiling
This project required some experimentation in order to get it to run. Since it is an app that runs on the website assessment tool we had to figure out whether or not we should try to get the app to run by itself or if we should attempt to get the assessment tool running and test it from there. After some trial and error we found out that the project builder Maven is required to compile the contrast finder. Once we found this out, we started to look into how to use Maven and found out in order to build this project we had to use the command "mvn test" from the command line in the Contrast Finder directory. After some time we got the Contrast Finder to build.

Conclusion
Ultimately we decided to choose this project because we got it to successfully build and it seems fairly simple to use, compared to STEM.