

CS486C – Senior Capstone Design in Computer Science

Project Description

Project Title: Graphical User Interface for massively multiplexed pathogen detection	
Sponsor Information:	Viacheslav Fofanov, Asst. Professor, SICCS Northern Arizona University Viacheslav.Fofanov@nau.edu

Introduction:

Pathogen detection is one of the fundamental challenges of modern biology. Being able to detect the presence of a disease-causing pathogen is the first step in formulating the treatment and containment strategy. Over the last 20 years, the scientific community has designed individual DNA-based molecular diagnostics tests for most diseases of interest. Often however, a simple detection is not enough. For example, during the Europe's 2011 pathogenic *E. coli* outbreak, detecting the bacterium contaminated foods was relatively easy. The tough part was to match the strains of bacteria to potential sources. Thousands of *E. coli* strains are circulating throughout Europe. Are contaminated foods coming from German lettuce? Spanish cucumbers? Something else entirely? Making a mistake can (and did) have profound consequences on the economies of entire countries.



Such strain-level detection requires hundreds to thousands of diagnostics tests (often one test per strain) to work in concert as part of a single pathogen screening panel. Each such test can adversely interact with others in the panel, leading to inaccurate results. Hundreds of thousands of potential interactions need to be monitored, evaluated, and optimized during the panel design process. In short, the computational expertise needed to design such diagnostic tools is not always found in clinicians tasked with outbreak investigations.

Problem Overview:

The Fofanov Lab at Northern Arizona University, has designed and implemented a software application that automatically reviews the possible interactions that arise during panel creations, scores the quality of potential panels and generally helps the user design a better diagnostic panel. This command-line software has already been extensively used by NAU's computer science and bioinformatics experts. However to make this software truly useful to the larger biology community, this software package needs an intuitive Graphical User Interface to help: (a) track the progress of the panel development process, (b) present the information that will enable **a biologist** to evaluate the quality of the panel, and finally (c) enable the user to guide panel redesign process.

Specifically, this project is centered around designing a Linux-based and/or Windows-based desktop application that includes the following features:

Basic (minimum viable) product must:

- Have a well-designed Graphical User Interface (GUI), validated on end-user biologists
- Have the ability to move between 3-4 steps or modules of the program
 - Load large datasets (1-2 TB)
 - Accept paths to data files
- Produce a csv text summary after each step/module
- Incorporate logic for inputs (limited input ranges and types)
- Monitor conditions for advancing/repeating steps
- Communicate with existing modules using .json files
- Implement and utilize default/config menu
- Be compile-able on iOS and Windows platforms
- Must have aesthetic appeal and be user friendly

A successful team will be able to go beyond the minimum features to add the following, which would make the product truly usable:

- Integrate rendered text summaries into GUI
- Mouse-over explanations for errors displayed during program use
- Drag-and-Drop data file selection

A gifted team will be able to include one or more of the following “stretch goals”:

- Progress Bar, that accurately calculates and shows time remaining in large computations.
- Interactive (possibly graphical) summaries
- Mouse-over hints for solutions to errors displayed during program use
- Query on-line databases for sequence data and retrieval (via ftp)

Knowledge, skills, and expertise required for this project:

- Knowledge of Linux-based application development, including GUI creation.
- Some skill in databases and/or storage, manipulation, and computation with large datasets.
- Skill in development and user-testing of GUIs including forms, drag-n-drop, videos, and other media elements.
- Interest or background in “big data” scientific computing will be useful.

Equipment Requirements:

- There should be no equipment or software required other than a development platform and software/tools freely available online.
- Testing datasets for development as well as technical expertise surrounding computational elements will be provided by the sponsor.

Software and other Deliverables:

- The GUI-based computational application, as outlined above, deployed on a hardware platform designated by client, and tested/refined with biologist end-users. Must include a complete and clear Administrator's Manual written for IT staff for installing, configuring and maintaining the software.
- An HTML-based online user's manual for biologist end-users, detailing how to complete typical computations/analyses using the product.
- Complete professionally-documented codebase, delivered both as a repository in GitHub, Bitbucket, or some other version control repository; and as a physical archive on a USB drive.
- A strong as-built report detailing the design and implementation of the product in a complete, clear and professional manner. This document should provide a strong basis for future development and growth of the product.