

The (Failed) Creation of a Language Change Simulator

The goal of this project is to create a software that simulates language change. Below is a list of steps I have to take towards completing the project. Of course, I was wayyyy too overly optimistic when I first envisioned the project...

Describing the language

In order to simulate language change, we must first describe the current state of the language. Therefore, our software needs to be able to read in and store the grammar of the language. The user should be able to input descriptions of an existing language, or they could use the software to create their own!

1 Phonemic inventory

The most basic component of a (spoken) language is a phoneme – the smallest unit of sound. Every language has a phonemic inventory, which is a collection of sounds that a speaker of that language perceives as distinctive. Each phoneme can be described using a feature matrix – a set of minimally necessary binary features that distinguishes it from all other phonemes in the language.

Thus, our software must be able to (1) store a phonemic inventory, (2) associate each phoneme to its set of features, and (3) allow the user to access phonemes via feature matrixes, and vice versa. As a bonus, we could also incorporate an algorithm that outputs a feature matrix given a list of feature values, and a system that allows the user to describe or create a phonetic-based writing system and associate each phoneme to an orthographical element.

2 Phonology

Every language has phonological rules that govern how phonemes are realized in speech. These rules are encoded using the same feature matrixes described above.

Thus, our software must be able to (1) allow the user to input and store phonological rules, and (2) correctly apply these rules to phonemic representations and yield the appropriate phonetic forms.

Time out...

So by then I realized that even getting that far is a bit of a stretch... And I was right...

An account of my unfortunate struggles

1 Creating a user interface

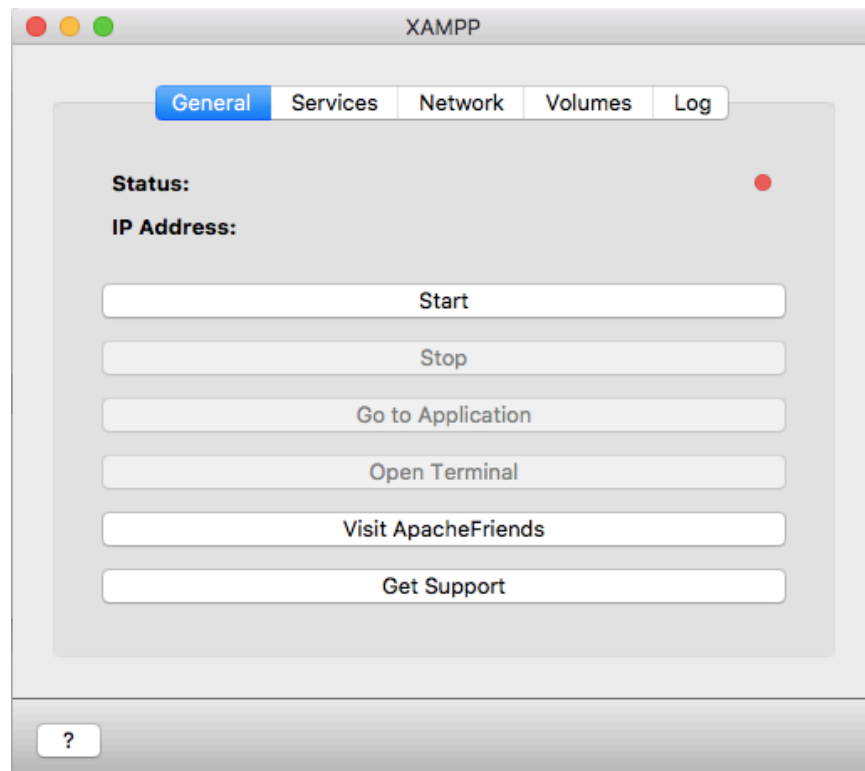
When I first started the project, I decided that the first thing I needed to do was to create a user interface. I found the *Qt5 C++ GUI Programming Cookbook* (URL:

https://culturalengineerassociation.weebly.com/uploads/8/6/7/7/86776910/qt5_c__gui_programming_cookbook.pdf) and followed instructions. I learned to use stylesheets, layout options, spacers, and various clicking reactions.

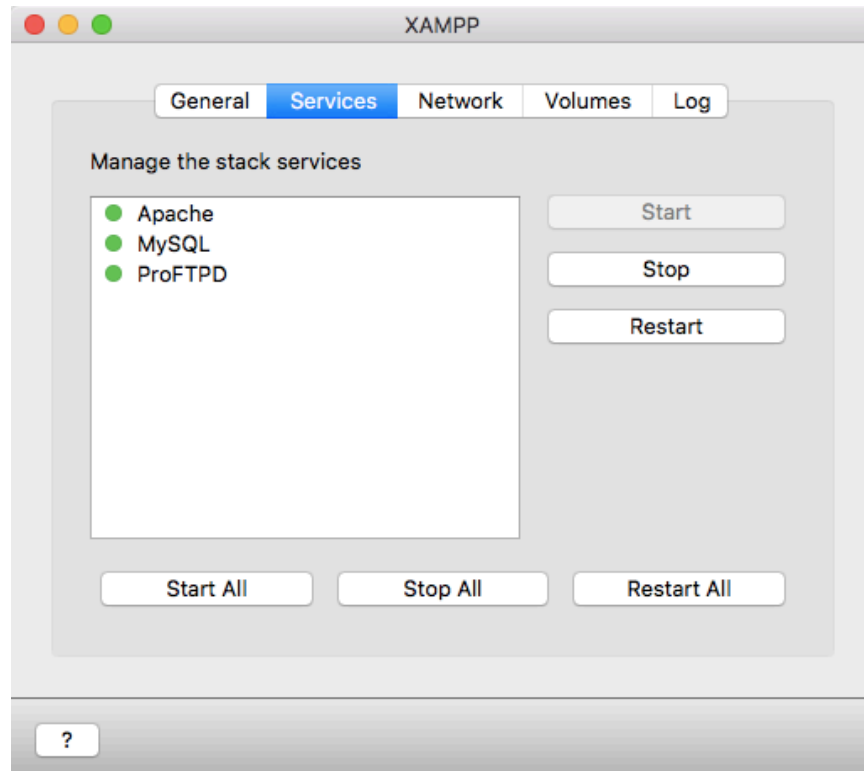
2 Setting up a database

Following the Qt Cookbook, I attempted to create and connect to a database. I downloaded XAMPP, which is a package that includes Apache and MySQL, which I need in order to create a database. Since the version of XAMPP that I downloaded has a very different user interface than the one described in the cookbook, I had to figure out how to make it work. Below is a list of steps (with screenshots!) that I took to set up a simple database for any unfortunate soul to follow:

- (1) Launch XAMPP and click “Start”



- (2) Click on “Services”. If the circles next to Apache and MySQL are red, click on each and click on start. If they are both green, you’re good to go!



- (3) Go back to “General” and click on “Go to Application”. It should take a page that looks like this:



XAMPP Apache + MariaDB + PHP + Perl

Welcome to XAMPP for 7.2.5-0

You have successfully installed XAMPP on this system! Now you can start using Apache, MariaDB, PHP and other components. You can find more info in the [FAQs](#) section or check the [HOW-TO Guides](#) for getting started with PHP applications.

XAMPP is meant only for development purposes. It has certain configuration settings that make it easy to develop locally but that are insecure if you want to have your installation accessible to others. If you want have your XAMPP accessible from the internet, make sure you understand the implications and you checked the [FAQs](#) to learn how to protect your site. Alternatively you can use [WAMP](#), [MAMP](#) or [LAMP](#) which are similar packages which are more suitable for production.

Start the XAMPP Control Panel to check the server status.

Community

XAMPP has been around for more than 10 years – there is a huge community behind it. You can get involved by joining our [Forums](#), adding yourself to the [Mailing List](#), and liking us on [Facebook](#), following our exploits on [Twitter](#), or adding us to your [Google+](#) circles.

Contribute to XAMPP translation at translate.apachefriends.org.

Can you help translate XAMPP for other community members? We need your help to translate XAMPP into different languages. We have set up a site, translate.apachefriends.org, where users can contribute translations.

- (4) Click “phpMyAdmin” at the top right corner of the page. You will most likely see this error message:

Access forbidden!

New XAMPP security concept:

Access to the requested directory is only available from the local network.

This setting can be configured in the file "httpd-xampp.conf".

If you think this is a server error, please contact the [webmaster](#).

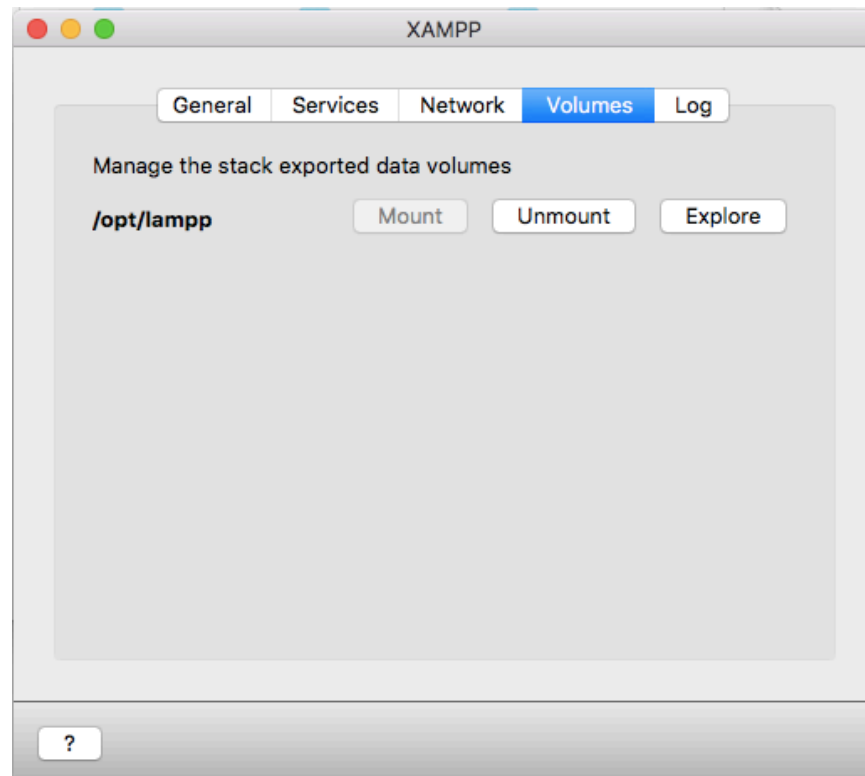
Error 403

192.168.64.3

Apache/2.4.33 (Unix) OpenSSL/1.0.2o PHP/7.2.6 mod_perl/2.0.8-dev Perl/v5.16.3

Note: It took me foreeeever to figure out how to fix this. You're welcome!

- (5) Go back to the XAMPP interface and click on “Volumes”. Then, click on “Mount”.



- (6) Click on “Explore”, and you’ll be taken to a Finder window that looks like this:

