

## MEMORANDUM

**To:** Patrick Peters

**From:** Steven Hill

**Subject:** Project Proposal

**Date:** June 10, 2013

---

### Summary

Easy-Terminal-Alternative (ETA) is a tool developed for the Center for Genome Research and Biocomputing (CGRB) to assist researchers in using the many command line tools available to them on the cluster (AKA cloud). ETA has made bridging the gap from a standard lab researcher to a computational scientist much easier, proving itself as a useful tool. It has the potential to grow to other institutions, but it lacks technical documentation required to install and maintain the software.

This project will give me the opportunity to use what I have learned in the last nine months to create meaningful and useful documentation for the software. Providing this documentation will provide me with an often-lacking skill in producing technical documentation for a large-scale application. Finally, it will be a meaningful contribution to the application and giving it the opportunity to grow its userbase.

### Introduction

The Center for Genome Research and Biocomputing at Oregon State recently added a new technology, Easy-Terminal-Alternative. Originally developed by Alexander Boyd, it was created to help less technologically skilled researchers use all of the computational tools available at the CGRB. Many people are not comfortable from the Linux command line, including Bioinformaticists. To solve this problem, Alex Boyd developed ETA. ETA is a web-driven tool in which users can create form wrappers for different computation tools located on the CGRB cluster (also known as a cloud). A researcher can simply look up the documentation for their application, insert the parameters to a form, and save this wrapper for future use by themselves and others.

From a technical standpoint, ETA is flawed. There is no technical documentation for the people expected to maintain the software. Maintainability is a key principle in software engineering and has hindered many other large scale projects in the past. As a direct result to this, other facilities that would like to install ETA cannot. ETA claims to be open source, but it is such a complex and unique application no one has bothered trying to extend it.

The proposed documentation will provide other facilities and developers means to install and extend ETA. The technology will expand and increase the production of Computational Scientists and grow the developer base for the technology. ETA has already proven to be a success at Oregon State University. Expanding the software to different institutions will be extremely valuable in helping researchers accomplish their Computation.

## **Proposed Program**

The technical documentation will be divided into four sections. Need and purpose, System Architecture, Installation, and Extension

### **Need and purpose**

. As described, this section will explain the need and purpose of ETA. It will also touch upon the use of the application and contain several screenshots of how the application is used. To expand on the need, it will also include an explanation of how researchers typically learned to use the command line to do computation, including the struggles that went along with that method.

### **Architecture**

Overview of the software's architecture and the design decisions that came with them. This will include UML diagrams and a list of the different technologies used. Architecture may also include various settings that are configured at compile-time.

### **Installation**

Instructions for installation. This step is extremely important as the software cannot be used if it cannot be installed. It will include instructions on installing the various dependencies. These dependencies are, Java 7, Google-web-toolkit, Tomcat 6, and a CentOS platform. It will also include how to configure the various settings such as authentication, and how to actually start the application.

### **Extension**

The extension portion will be used to describe the different modules of the application and the documentation with how to implement a new extension of these modules. These include different Authentication services, Batch services, User services, and modifying the user interface. It will also include method-by-method documentation for these services.

## **Qualifications and Experience**

I have developed and debugged ETA for the past nine months. Through these experiences I have learned to install ETA and extend certain portions of the software. Only two people have ever worked on the application, and I am one of them.

## **Time-Table**

Table 1: Time Table

Date	Task
May 17	Gather Interview with Researchers and Alex Boyd
May 20	Complete UML Diagrams
May 22	Progress Report, Sample, and Outline
May 29	First Draft Finished
June 2	Oral Presentation
June 6	Try document on programmer who has not seen ETA
June 10	Final Document

## Works Cited

- Ambra Molesini, Alessandro Garcia, Christina von Flach Garcia Chavez, Thais Vasconcelos Batista, Stability assessment of aspect-oriented software architectures: A quantitative study, *Journal of Systems and Software*, Volume 83, Issue 5, May 2010, Pages 711-722, ISSN 0164-1212, 10.1016/j.jss.2009.05.022.  
<http://www.sciencedirect.com/science/article/pii/S0164121209001162>. ETA is developed using an aspect-oriented design. A more detailed description of these patterns are important for the architecture section.
- "Apache Tomcat 6.0." *Tomcat Documentation Index*. 9 Oct. 2012. Apache Foundation. 13 May 2013 <http://tomcat.apache.org/tomcat-6.0-doc/>. The webserver the application lives on is Tomcat. Installation relies on the knowledge of Tomcat.
- "Google Web Toolkit." *Developer's Guide*. 26 Oct. 2012. Google. 13 May 2013 <https://developers.google.com/web-toolkit/doc/latest/DevGuide>. The entire user interface was developed with the Google Web Toolkit. Referencing these documents is a must.
- "Java Platform SE 7." *Java Platform SE 7*. 28 July 2011. Oracle Inc. 13 May 2013 <http://docs.oracle.com/javase/7/docs/api/>. The Java API is the official documentation of Java and is the language used by ETA. Many of the features are already described in these documents.
- Schroder, Carla. "Linux Bug # 1: Bad Documentation" *The Many Faces of Documentation*. N.p., 17 Nov. 2009. Web. 13 May 2013.  
<http://www.linuxplanet.com/linuxplanet/reports/6904/1/>. Carla Schroder is the author of Linux Cookbook and has been working with Linux since 1997.