

|  |
| --- |
| **CSCI Department – CS Capstone I (P445) – Feasibility report** |
| PHYLOGENETICS DATA INTERFACE |
|  |
| **Kevin Florian | Joshua Rowe | Eric Wright** |
| Team “Armadillo Island Software” |

|  |
| --- |
|  |

PRODUCT

The proposed project will be a phylogenetic analysis program. The program will model the functionality of programs like PAUP and McClade or enhance the productivity of using such programs by providing an interface for entering data more rapidly.. This may include calculating phylogenies and graphical displaying the results in tree form. There will also be support to export data entered into the program out to what are called Nexus files. These are the files that many of these phylogenetic programs use to do their calculations. It will use several techniques that are used to calculate phylogenies when given taxon data if that is the main purpose of the program.. The initial approach will be to implement functionality that is central to what Dr. Taylor and biology students here at IUS need to perform their calculations. In this respect, it is very feasible that performance and functionality will meet the required specifications of such a program.

 TECHNICAL FEASIBILITY

The proposed system is modeling a subset of the functionality of more widely used programs such as PAUP and McClade. This being the case, the specification of the program is a lot less strenuous than that of completely rewriting all of the functionality of these phylogenetic programs. In this regard, the likely-hood of the program be able to perform to all of the user’s needs is very high. Dr. Taylor has made it clear that having an application that increases the productivity of his work sad possibly the work of his students could be even more valuable than having an undated version of a program that actually does phylogenetic analysis, so we will probably end up going that route for that and other reasons.

SOCIAL FEASIBILITY

This system is unique as there has not been any system created like it in over 20 years, thus it’s time to update. This system would almost exclusively be utilized by biologists, and biology students, who are studying phylogenetics. This system would not demean these expert’s skills as knowledge on the subject is necessary to interpret the results. However, this updated system would definitely help simplify and speed the process of analyzing relations dramatically. As the current process is very timely, this system will likely make this aspect of Phylogeny easier. When the time comes to introduce changes, a face-to-face meeting with a phylogenist will be necessary to help pinpoint the most accurate algorithms.

ECONOMIC FEASIBILITY

The only cost for this system for biologists would be to download and learn how to use said system. Once the user knows how to utilize this, this system does nothing but save. This system will reduce a great amount of time and effort that currently exists in phylogeny. The time no longer spent on the current, drawn out process, could then be used to add to further research.

MARKET RESEARCH AND ALTERNATIVE SOLUTIONS

Within the many disciplines of bioinformatics, phylogenetics seeks to study the evolutionary history among groups of organisms by analyzing molecular data sequences. As such, it has become highly informative to other related disciplines (e.g. taxonomy, phylogenomics) and in areas as diverse as ecology and forensic studies.    The need for phylogenetic tree modeling and interpretation has seen several software submissions within the tech forum of Biology in the past couple decades. Many applications are method- and/or data-specific (e.g., distance matrix only, nucleotide only), such as fastDNAml and PhyML. Some applications are more lightweight in function and scope, offering only basic visualizations and data processing (e.g., QuickTree, TreeGen). Lastly, a handful of software packages perform a myriad of bioinformatics functions beyond just Phylogenetic tree analysis, (e.g., MEGA, BaseSpace MetaPhlAn). A list of computational phylogenetics software programs is supplied on Wikipedia with details on those mentioned (https://en.wikipedia.org/wiki/List\_of\_phylogenetics\_software  (Links to an external site.) ).    Amongst them, PAUP (Phylogenetic Analysis Using Parsimony) became the most widely used for phylogenetic evolutionary tree inferencing. Currently, the versions of PAUP available are useful and still functional, but stuck in outmoded presentation and old methods of data formation (i.e., NEXUS files created manually). For these reasons – and given the context of PAUP’s user-base in education – the need for a modern, updated, user-friendly data interface for current phylo-tree inference software is clear.    The creation of a data interface (nexus file manipulation) on a mobile-friendly platform with an emphasis on modern UI will set our niche package apart from the rest. With efficient PAUP integration, our interface will serve as a powerful phylogenetics tool in the realm of education where students will benefit from its mobility, clarity, and efficiency.    Due to the already established algorithms and analysis methods in this branch of bioinformatics, and given the traditional common data format of NEXUS files, there really are no alternative solutions to this very particular vacancy/pinch in the process. Granted, more invasive/disruptive alternatives can be imagined, this niche involving phylogenetic data-handling is quite precise.

PROJECT RISKS

The project has several risks associated with it as any software does. Firstly, there is a risk of a new piece of software cropping up that solves the specific problem that we are aiming to solve via our implementation. If a piece of software were to arise, and be produced professionally, then our implementation, might seem inadequate for the purposes of the IUS biology department’s uses. The relative risk of such an event happening is very low, though, as the use of phylogenetic analysis programs is very domain specific and might not yield the profits that a professional development studio or individual would look for in trying to produce a piece of software.  Second, the risk of the scope of the project being more that what a group of our size can handle is also a possibility. It is reasonable to think that the risk of this is also low due to the fact that we are going to use a dynamic development strategy, and we plan to start with simple goal and expand outward if we realize that the goal is easily obtainable.  There are also other risks, but none that endanger the project as long as all team members contribute and we remain on time.