

The class project

We have begun a class project that we're building upon each week in which we are populating a Chado database with selected attributes from the annotation of a single bacterial genome project. As we go through each week's lecture, we'll continue building an interface that provides methods to browse the annotated genes in the organism as well as a search interface. The project will involve using the following technologies and methods covered in class:

- The UNIX OS and filesystem
- Placement/organization of files within a web application
- Python CGI programming
- CGI and HTML templates
- HTML5 document markup
- Relational database schema design
- MySQL
- Using mysql.connector python module to connect to MySQL
- Limited page styling with CSS
- Javascript and JQuery for client-side interaction

This will be built in an iterative fashion. We'll add some parts, then change them, then replace them completely with other methods. You should be sure to focus on an understanding of how all these portions connect to yield a final product because your Final Project will need to utilize them as well.

The final project

Description:

You'll need to propose your own bioinformatics-related project that uses the technologies listed above. It should be biologically relevant and can either be a data display tool or an analysis tool. A database needs to be utilized here, so if you decide to create a tool that performs an online analysis and doesn't rely on data stored in a database you can do something like storing the results of any submitted analyses. You can use any schema you wish, but you need to justify the schema you've chosen. The principles of content separation should be employed, so don't do things like print HTML directly within your CGI scripts, and don't embed Javascript and style information into your HTML files, whenever possible. (This also means you shouldn't use markup that is presentational only, such as loading your code with `
` tags.) I'm more concerned about it being functional. If you want to make it pretty, use CSS.

If your project requires python modules or any other software to be installed just send me a message and I'll install it. Just keep in mind that all students in this course share the same server, so proposals to build an advanced protein folding simulator, however ambitious, aren't practical given these shared resources.

The Class Project should give you a sense of the size and scope your own project should be. The purpose of this exercise is to show that you've learned how to implement a web-based bioinformatics analysis application using a modern technology stack.

Proposal:

You'll need to write up a description of your project idea. It should be technically detailed and describe what potential users would be able to do, and what would happen on the server side with respect to the technologies listed above. There is no formal format here – I just want a description of your project with what functionalities it will provide in a level of detail that addresses the technical requirements above. Most proposals should approach a page in length.

Submit your proposals in the assignment section of Blackboard, and I'll read over them and possibly suggest additions or clarifications before approving your project. This will help ensure that your proposal meets the requirements for full credit.

Deliverables:

April 15th, 11:59 PM EST: Project proposals are due on Blackboard

May 15th, 5:00 PM EST: By this time you must have all of your code deployed on the class server as well as a tarball of your project code uploaded to Blackboard. This must contain a README file with simple documentation on how to use your application.

Ideas:

There are different types of projects that are commonly chosen. The first one is a data mining tool that allows users to search and browse otherwise less-accessible data. This would require you to find and download the source data, create a data model for it in your database, parse and load the source files into it, then create an interface that allows the user to interrogate it. This might include data sources like:

- annotation collections
- ontologies (obofoundry.org)
- custom biological databases
 - repeat libraries
 - enzyme collections
 - rRNA databases
 - restriction enzymes
 - etc.

The second sort of project allows the user to perform an analysis on the server. Typically, they enter a sequence in a form element and your code does an analysis and reports the results. This can be your own custom analysis code or the execution of a tool that runs relatively quickly. This is especially better if you can take the plain text output of a tool and make it visually more pleasing or better summarized. I'm hesitant to suggest tools here because any time I suggest a few all the projects I get back are just those few tools. Use google, search something like "bioinformatics tool" and find something that executes quickly.