CSSE 403 Final Project

In the CSSE403 final project your **2/3-person project group** will:

1. Pick an “interesting” language, not taught in the class
2. Do a 1-period long class presentation on that language, including a “graded” programming activity
3. Build a project in that language that highlights the strengths/unusual features of the language you’ve chosen

This project is worth 19% of your overall course grade.

## Choosing Your Language

You can choose and programming language you want, with these restrictions:

1. It should be unusual in some way. Ruby, Java, C# - these are all very mainstream languages with features that (for the most part) are not very different from each other and would not make a good choice. One place to start looking is the other languages in your textbook (or maybe its sequel, *Seven More Programming Languages in 7 Weeks*) – but there are many interesting languages out there.
2. It should not be a joke language (e.g. [Whitespace](http://en.wikipedia.org/wiki/Whitespace_(programming_language))) – there should be a community of people who genuinely think this language is a good idea at least for some purposes.
3. It should be unique in the class (first come, first served).
4. I must approve it.

When you have picked a language you should get it informally approved by me. Then go on to do the official proposal (described at the bottom of the document).

# The Class Presentation

You will do a class presentation on your language. This should be a full 50-minute class period, and every team member should in some way participate. The presentation should be well-rehearsed (with notes, sides, or whatever you need), clear, and interesting.

Because the class will only get a 1-day introduction you should try your best not to just give them a “hello world” talk but introduce them to some specific and interesting feature of the language. It is understandable that they will not get a full flavor of the language in 50 minutes but try to do something memorable that will give a real feel for the language’s strengths/uniqueness.

You do not have to demo your project as part of the class presentation – do it only if it is educationally useful.

## The Class Presentation Activity

Because one of the premises of this class is that you can’t really understand a language without programming in it, your class must have a programming activity. Here are some guidelines:

1. You should send out an email before your class too [csse403@rose-hulman.edu](mailto:csse403@rose-hulman.edu) letting people know what they should install to work on the activity.
2. It should be a little longer than hello world and provide at least a small challenge.
3. You should not give the solution to the whole activity in class. You can have some parts where the solution is revealed, but for some parts the class should be left to their own devices.
4. You can give students in class time to work on the activity – but don’t make the whole class just work time.
5. You can make the activity slightly longer than can be accomplished in class, so that folks work on it as homework. But be very sparing with this – no more than 40 minutes outside of class.
6. You should have enough resources (notes, slides, links to online resources, and activity guidelines) that if a student misses class they can plausibly solve it working independently (again, mail these to the class).
7. You will collect the activity and grade it – it will be worth a 10 point small assignment. You can be generous but you should have some standards.

## Grading the Class Presentation (50 points)

|  |  |
| --- | --- |
| Prepared to start. Class emails were sent with sufficient notice, every group member is present, notes/slides/whatever are ready to go as class begins. | 10 points |
| Polished and rehearsed presentation. Thought clearly went into building the materials. People know who is talking when and what should happen next. Fills the time correctly. | 10 points |
| Presentation is clear and interesting. | 10 points |
| Activity is well designed, interesting, and at the appropriate level of difficulty. | 10 points |
| Activity is graded on time and sent to me. | 10 points |

## When is the Class Activity Due?

Your team will be assigned a particular class period randomly.

# The Programming Project (50 points)

You also will do a medium/large scale project if your choice in your language of choice. The project should be a task for which your chosen language is well suited. This project should be at minimum as large as the “medium projects” you have done in class – it could be a little larger. Every member must participate – so you should give some thought to how to divide things up so that everyone is doing something interesting.

## The Programming Project Proposal

You will formally propose a programming project in writing to me. You should:

1. Briefly describe the project in general terms and why your language is well suited for it
2. Give a breakdown of the 50 points of the project into functionality areas like I do on my assignments. Make sure it is clear what you require for at least a majority of the sections – this is going to be my approximate “rubric”. You can have one “random cool ideas as we decide later” section (not required).
3. Note who is going to work on what sections of the project – pairs of people are fine if you are actually pair programming, not if you’re just too lazy to decide. This is not set in stone once you start, but I want to see you’ve at least got a plausible division of labor that ensures everybody does something.
4. The proposal due date is on the schedule.

## Grading the Project

I will probably stick approximately to the rubric you have given me. The whole team will get the same grade unless there’s some inter-team conflict or someone does not do their work. If there is a conflict:

1. Address it with me before it delays the overall success of the project
2. Address it in a professional manner that does not involve ignoring team meetings/emails/phone calls or going off in a huff and implementing the whole project yourself.

## Submitting the Project

You will use source control for this project. I can make you a repo on the class SVN, or you can use some other source control and add me. Do not use something without change history (e.g. a shared dropbox). You are required to commit code under your own name, using your own account. In case of a conflict, I will look to source control to determine who wrote what code when.

You will also demo the project for me in person. **Demo day is usually the first day of finals but check the schedule. You project must be complete by your demo time.**