The goal of this project is to (again) work in a team of two to learn a new programming language and use it to develop an application. This project has four deliverables: (1) a project proposal; (2) a status update; (3) a program written in the language; and (4) a video presentation.

**Project Proposal (Due Thursday, March 24th).** The proposal involves the following steps:

1. Select a language to learn (other than Haskell and Prolog, which we’ll cover in class). Pick a language that has functional features (e.g., Java, C++, Python, JavaScript, Erlang) or that is a functional programming language (e.g., F#, Scala, ML, OCaml). If you pick a language that isn’t functional, you will need to focus on the functional features of the language. The goal is to learn something new.

2. Find online resources that you will start looking at to help you learn the language.

3. Brainstorm a programming project related to one of the following themes:
   - Environmental Sustainability
   - Access to Healthcare
   - STEM Education
   - Homelessness and Housing
   - Civic Engagement

When brainstorming your app to develop, identify:

(a) who the potential users are

(b) what the needs of the users are (i.e., what problems or “pain points” do they have) with respect to the theme

(c) the value that the app would have for these users (how it solves their “pain points”)

(d) the app features that would be needed to provide this value

Your job will be to develop a prototype (i.e., not all features need to be implemented and/or implemented fully) of this app in the language you selected. Your prototype should be a proof-of-concept of your app so that others can get a sense for how your app would function. Your prototype must be implemented such that it uses all of the main constructs of the programming language you choose.

Hand in your proposal with the language you are going to learn, a brief description of why you selected the language, a description of your app (the theme, the users and pain points, and the features), a brief sketch of what the prototype will consist of, and the initial language resources you will be looking at.

**Status Update (Due Thursday, April 14th).** Turn in a one-page description of the progress you have made. By this point, you should have a clear understanding of the language (possibly with a few things still left to learn) and a good start on your app. Your update must include:

1. A summary of how far you are, including what resources you have finished

2. A list of what you are still trying to figure out, if anything

3. A description of your prototype app (which should be more concrete than in your proposal)

4. What features you have implemented and the overall design

**Video Presentation and Program (Due Thursday, May 5).** You must hand in your finished program and a video presentation on your project results. Your presentation must be at most 10 minutes long and include:
1. A brief overview of the history and main features of the language you selected
2. The main programming constructs supported by the language with examples
3. A description and demo of the program you wrote
4. Highlights of your program’s design (in terms of how you implemented it)
5. A summary of your overall impressions of the language

Your video must have both audio and visual components (e.g., slides). One way to accomplish this is to design a Powerpoint presentation and create a video from it. You can also use a Kahn-academy style presentation or another approach of your choosing. Upload your video presentation to YouTube and provide me with a link. You must also turn in hardcopy of the source code for your application as well as a link to the source code with instructions on how to run it (e.g., via GitHub or on ada).