

CSCE 120 – Learning to Code Group Project

Overview

A substantial amount of your grade in this course will be based on your group project. Your group should consist of 3-4 members. The nature of the project is up to your group to decide, but must be a substantial application that may be:

- A basic web application
- An interactive “infographic” or data visualization

You should choose a project in which you have a personal interest. It could relate to a hobby, a personal interest (sports, politics, etc.) or you may consider adapting a project or topic from one of your other courses. Though the choice is ultimately yours, below we have provided several examples and ideas.

You are expected to work closely with each member of your group. The project should reflect an equal contribution from each member. You will be expected to present your final project to the entire class for group evaluation.

Deliverables & Timeline

There will be several project workdays throughout the semester. See the course schedule for specific dates.

- Workday 1: Prior to workday one you must have identified an idea, topic or data set that is of interest to you. On workday 1, you will informally share your idea/topic/data set with your colleagues. Based on this, it is expected that you will start to form groups. It could be that someone else’s idea(s) interest you more or that some ideas are similar or complement each other and would make a better project together.
- Workday 2: Prior to workday two you must have formed your group and decided on the scope of your project. Workday 2 will give you a chance to work out the scheduling details and plan for how you will go about implementing the project. Beyond workday 2, you will be expected to meet outside class to work on your project.
- The Final Exam time will be used for each group to give an informal demonstration/presentation of their final projects. All project code, artifacts, etc. must be handed in electronically through webhandin. Note that your project may be included in a project showcase for future students.

Examples

Below are some examples of various data visualization and small apps that should serve as inspiration for your projects. Each of these infographics and visualizations

use publicly available data and open JavaScript libraries. A successful student in this course should be able to accomplish something similar to these.

- Visualization of spending habits versus education level:
<http://www.retale.com/info/degrees-of-spending/>
- “World Cup of Everything Else” - Wall Street Journal infographic comparing countries based on various statistics:
<http://graphics.wsj.com/documents/WORLDCUPTOEE/>
- Tracking 40 years of college majors – an NPR infographic:
<http://www.npr.org/blogs/money/2014/05/09/310114739/whats-your-major-four-decades-of-college-degrees-in-1-graph>
- Where Do College Graduates Work? A d3js-based infographic mapping degrees to occupations:
<http://www.census.gov/dataviz/visualizations/stem/stem-html/>
- Census Data Visualizations – A gallery of various data visualizations based on US Government Census data:
<http://www.census.gov/dataviz/>
- Github Data Visualization – An interactive visualization of data from Github, a public code repository and collaboration website:
<http://danielvdende.com/gdc2014/>
- GitHub – Interactive data visualization and relationship mapping based on historic data from Github. Shows how programming languages become more or less popular over time:
<http://github.info/>
- Wine, Beer or Spirits? – A visualization of global alcohol consumption based on World Health Organization data:
<http://ghostinthedata.com/alcoholmap/>
- Data Is Beautiful – A forum on reddit that highlights data visualization:
<http://www.reddit.com/r/dataisbeautiful>

Application Ideas

- Create a post-it note style web app that allows a user to create as many small, post-it notes as they want. They can write notes in each one, drag each one around on the page, delete notes (perhaps by dragging to a trash can), and save their notes (at least the content if not the positioning) to localStorage which is loaded when they visit the page again.
- Create an application that displays (multiple) RSS feeds. It should allow a user to add and remove feeds from their page (via urls). Google search could be integrated to find other interesting RSS feeds. Feed lists could be saved via localStorage. Note that RSS feeds are usually formatted as XML, but Google has a public API that converts feeds to JSON for you. An example:
<https://ajax.googleapis.com/ajax/services/feed/load?v=2.0&q=http://www.huskers.com/rss.dbml&num=20>
- Get an API Key to ESPN.com (see <http://developer.espn.com/docs>) and write an app that aggregates various sports data together into a custom page.

Examples may include graphing win-percentages for any team with a sliding window (last 10 games).

- Write an application that will allow users to enter information on multiple loans that they have (balance, monthly payments, interest rates, etc.) and an amount of money that they can apply to pay extra each month. Visualize the amount of interest and principle each month and give the user a comparison of various repayment strategies (snow-ball vs stacking/avalanche). Inspired by the following app: <http://unbury.us/>
- Create a simple photo-effects app that allows users to apply a variety of customizable photo effects using HTML5's canvas functionality
- Write a grocery list app. The app could pull in nutrition data or pricing data if available.
- Pull data from a website that ranks things (IMDB, Amazon, etc.) and rerank them using a more robust statistical method such as Wilson's confidence measure (see: <http://www.evanmiller.org/how-not-to-sort-by-average-rating.html>). Visualize the different rankings in some manner.
- Create a "mashup" of public data feeds from various websites. Some available APIs: Reddit: <http://www.reddit.com/dev/api>
 - ESPN: <http://developer.espn.com/docs>
 - Steam Store API:
https://developer.valvesoftware.com/wiki/Steam_Web_API
 - APIs from Iceland: <http://docs.apis.is/>
- Exercise: pull data from a website that does ranking "wrong" and do it right (using a statistic such as Wilson's confidence measure; see: <http://www.evanmiller.org/how-not-to-sort-by-average-rating.html>)

Libraries, Resources, and Other Inspirations

- Information Is Beautiful – a collection of data visualizations and infographics on a variety of datasets:
<http://www.informationisbeautiful.net/>
- jQuery Charts Plugin – a jQuery/jQuery UI plugin that allows you to graph data using a variety of chart types:
<http://canvasjs.com/jquery-charts/>
- Data Driven Documents – a library of data visualization functionality:
<http://d3js.org/>
- Google APIs:
<https://developers.google.com/apis-explorer/>