Task: Concentration Track

- Recall the MADLIB lab
  - That was all front end
  - Your job is to make it a complete app
  - You can use your previous lab, one of the examples posted, or start over
  - You can do the work using templates or in the front end using AJAX

- Front end:
  - Home page (description and start button)
  - MadLib page (input for current mad lib)
  - Output page (output for current mad lib)
    - Can be same as MadLib page
    - Button to generate a new mad lib
Task: Designer Track

• Can pair with concentrators
  • You are in charge of the html (template), look and feel, etc.
  • Consider: randomizing input order, coloring inserted words, ...

• Alternatively
  • Design a whole web site for the game
  • Possible extended features
    • Log in (track which mad libs have been done to ensure a new one)
    • Sharing mad libs (show other peoples answers after your answer)
    • Rating other people’s mad libs
    • ...
  • Create some mad libs for everyone to use (in class only)
Implementation Notes

• List the tasks.
  • Decide where each task is implemented
• Decide on the URLs to be used
  • For each request
• Create a simple node.js/express server
  • Extend for the various requests
• If you want to make the result available publically
  • Send me all the necessary files in a bundle
  • spr+17lab04@cs.brown.edu
  • I’ll try to put it up on a brown vm
Next Time

• Databases

• Homework:
  • Consider the beiberfeed data. Suppose you wanted to store it permanently. List the data that you would want to save and the implicit (or explicit) relationships among that data.