Something in professor's ppt

Lets Beat Google!

- Stage 1 (30%+): Page Ranking
  - Given a set of keywords and URLs
  - Rank the URLs based on their score
  - Define a score formula based on keyword appearances
  - For each URL (a web page), return its rank, score, and the count on appearance of each keyword

In HW it must be like: HW3+HW4

- The content of HW4 is responsible for maintaining the list of keywords,
  while HW3 is responsible for calculating the frequency of these keywords appearing in webpages.

- You should integrate these two assignments and develop some classes that can evaluate the score of a webpage.

Something in professor's ppt

Lets Beat Google!

- Stage 2 (50%+) Site Ranking
  - Multiple level keyword search
  - Given a set of Web sites (URLs) and Keywords
  - Rank the Web sites with their keyword appearances (including **all its sub URLs)**
  - Define a score formula based on keyword appearances in the URL and all its sub URLs
  - For each URL (a web site), return its rank, score, and a tree structure for its sub URLs along with the number of appearance of each keyword in each node

In HW it must be like: HW6+HW8+(sorting)

- The content of HW8 can assist you in analyzing HTML tags
  (primarily used for fetching URLs of subpages) ,
  while HW6 includes building the entire website's tree structure and
  calculating the score for this tree.
- You should combine these two assignments to develop a functionality that, based on the search results for the user input keyword, can build the tree structure and calculate scores for each (or several) website(s), finally sorting according their scores.

Something in professor’s ppt

Lets Beat Google!

- Stage 3 (70%+) Refine the rank of Google
  - Given a set of Keywords (No URLs)
  - Use **search engines** to find potential URLs
  - Apply the ranking on Stage 2 to these Web sites

In HW it must be like: Just like in Stage 2.

- This stage is more tricky.

  Google's search result may not satisfy our requirements,

  and some potentially important websites may not appear in the direct search results using the user input keyword.

- Therefore, it might be necessary to prepare an additional set of keywords to uncover these search results.

Something in professor’s ppt

Lets Beat Google!

- Stage 4 (80%+) Semantics Analysis
  - Derive **relative keywords** from the discovered Web sites
  - Iteratively do the same analysis on Stage 3

In HW it must be like: HW10

- Stage 4 can be roughly understood as how to generate new keywords related to your topic from the originally found websites.
- The default keywords you set may not be the best, so you can employ techniques like LCS or identifying the most frequently occurring words to add new keywords in your keyword list.

- In this stage, you can use the techniques learned in class or used in HW10. Just implement them for something you collect from websites.

Something in professor’s ppt

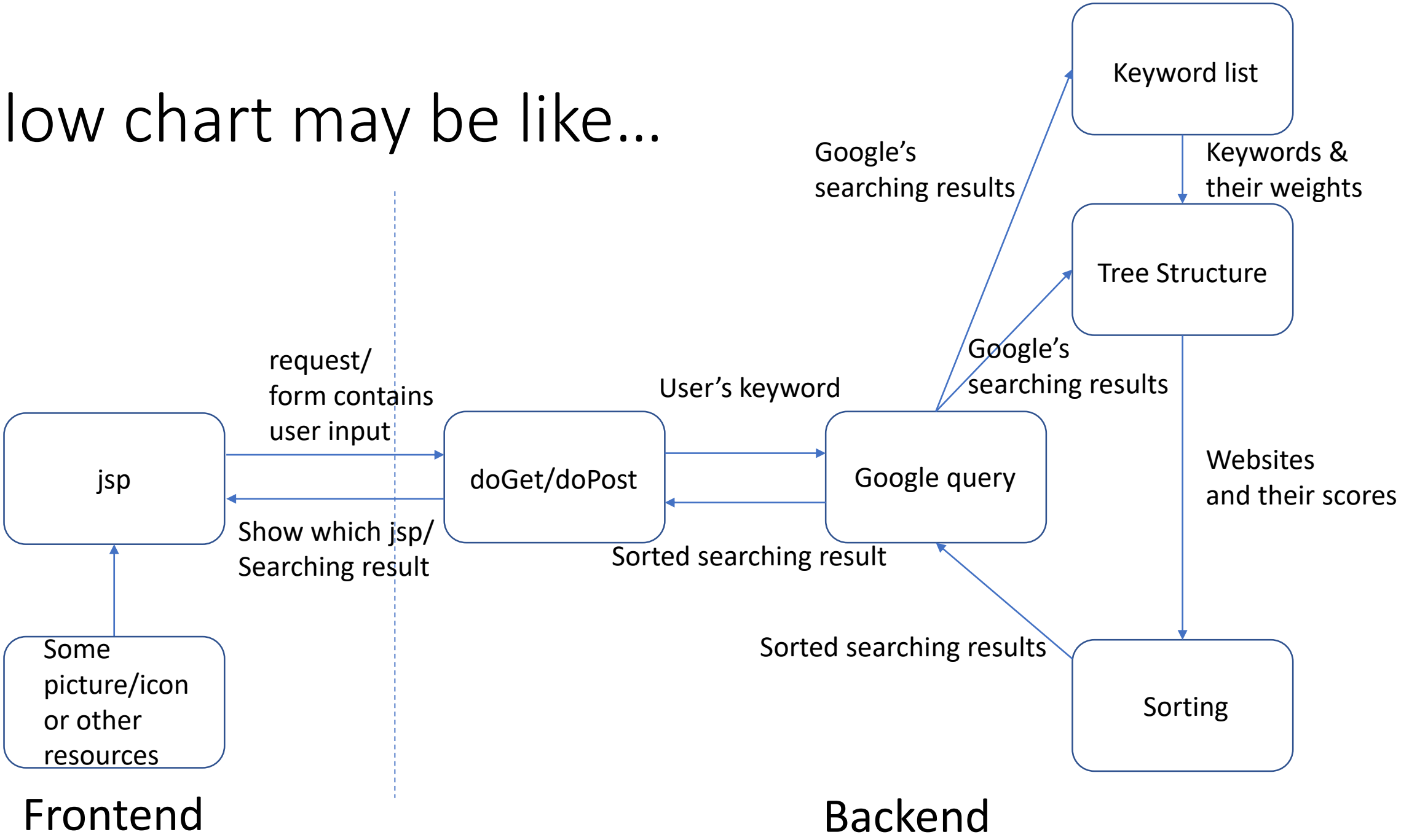
- Stage 5 (90%+) Publish Your Work Online
  - Build a web site/service for your searching engine

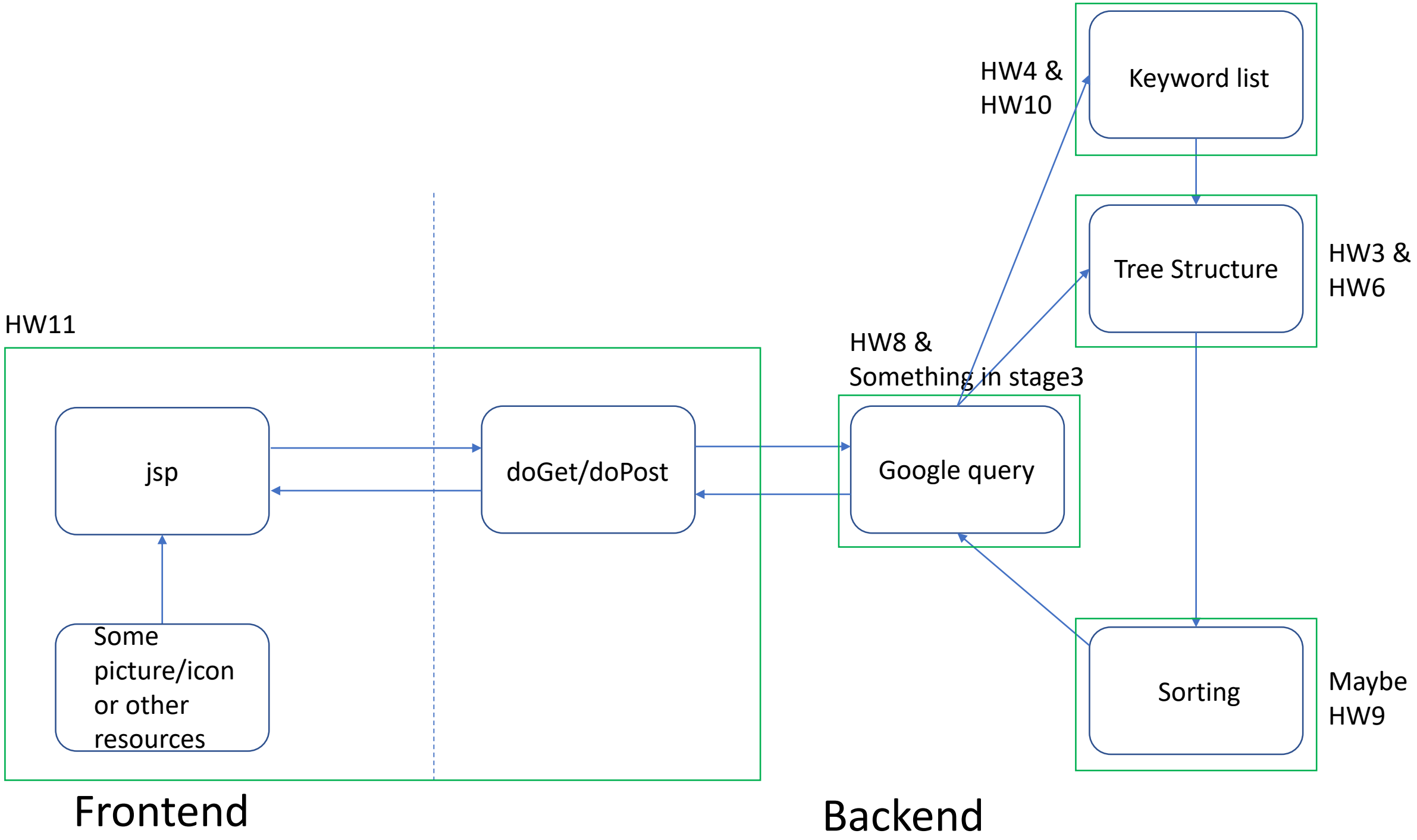
In HW it must be like: HW11

- This stage is similar to the Testproject page in HW11.
- If you can run Testproject page and successfully show search results, then you are only a step away from completing your team project by integrating the content from the earlier stages with this one.

- Although I understand there might still be various environmental issues with the web part =)

Flow chart may be like…




HW4 &
HW10
Keyword list
Tree Structure
HW3 &
HW6
HW11
HW8 &
Something in stage3
jsp
doGet/doPost
Google query
Some
picture/icon
or other
resources
Sorting
Maybe
HW9
Frontend
Backend