Technical Challenge
Principal Front End Software Engineer

Instructions

Please complete the following assignment and be ready to demonstrate how it works, show any code you built and discuss any challenges you faced during your next meeting with the interviewing team.

Some aspects of this exercise are intentionally vague and left up to you. You may use whatever technologies and libraries you wish. We are primarily assessing: code clarity, application architecture, UI + UX decisions.

We are happy to answer any questions you may have.

Create a short document explaining the architecture and any assumptions you made.

DNA Secondary Structure Representation

Create a web page which, given a DNA sequence and its 2D structure in [dot-bracket notation](http://biojs.io/d/drawrnajs) (DBN), generates an interactive 2D representation of the molecule. You should display all representations on the page. For an example of something similar to what you're building (but with more features), see: http://biojs.io/d/drawrnajs

The DNA Sequence will contain standard A, C, T, G bases, possibly N (any base), in the 5'-to-3' direction.

The 2D notation will contain only dots and parentheses (e.g. no pseudo knots).

Requirements
Each base of the sequence should be colored.

The DBN should be visible.

The 2D visual representation should color and label each base, annotate 5' and 3' end, and visually differentiate edges for phosphate backbone and base-pair complementarity

There should be ways to configure the following aspects of this view:
- colors of each base
- size of the base representation
- font for the label
- line width

The graph layout does not need to be deterministic. It should be close to the planar graph layout as shown in the example below. The D3 force layout is a good basis, but you are welcome to use other techniques.

Interactivity
The user can drag bases in the graph to modify the layout.
When hovering a base in either the sequence or graph, the corresponding base should be highlighted in the other view.
The user may create their own connections, by connecting two unpaired and complementary bases (e.g. C and G, or A and T). When making a connection, the DBN should update. The user experience around this interaction is up to you.

Sharing
In order to share a particular view with a colleague, the user should be able to obtain a customized URL that includes the state of the views. Just navigating to that URL would open the exact same view.

Example
Sequence (100 bases):
TTGGGGGGACTGGGGCTCCCATTCGTTGCCTTTATAAATCCTTGCAAGCCAATTAACAGGTTGGTGAGGG
GCTTGGGTGAAAAGGTGCTTAAGACTCCGT

DBN (100 bases):
...(((((.(....).)))))........((((.....((..(.((((((..(((.
((...)).))))..))))))).).)))))))...............

2D representation (from nupack.org):