

# Beta: Progress Report

Mike Fang  
mf647

Andrei Shpilenok  
ais76

December 6, 2020

## Vision

Our vision of unifying content and styling natively has not changed with this sprint. Our goal is to develop a fully fleshed out language for the web which brings together CSS and HTML and absolves javascript of the responsibility of having to do everything. We also aim to improve styling in the web by integrating constraints with our language. We envision the project to consist of two main parts:

1. Designing and Parsing the Language
2. Creating a Dummy Browser to Display the Result of the Language

After creating a dummy, end-to-end implementation, we realized we will not be able realize a fully-fleshed out union of CSS and HTML, as we will not have the time to implement all features. However, we believe we will still be able to implement a language which will be able to structure and describe content in a more concise way.

## Progress

This sprint we have completed an end-to-end, tracer-bullet implementation. The user is now able to draw nodes using the language. The nodes show up on the canvas as green rectangles.

The way this works is as follows:

1. Parsed Source Code is transformed into a Stylesheet and DOM
2. DOM and Stylesheet are combined into the Style Tree (Now we know what nodes we have to paint and their corresponding styles)
3. Walk the Style Tree and each nodes constraint styles to create a Constraint Variable Pool which is a map from Style Tree Node IDs to attribute names to constraint variables

4. Walk the Style Tree again and gather all constraint objects into the constraint solver
5. Construct the Render Tree which consists of identical nodes to the Style Tree, but they now store the direct values of attributes which are retrieved from the solver
6. The Render Tree is then passed to the Scene module to be rendered

## **Activity**

### **Andrei**

1. Wrote the Graphics Portion of the Blog Post
2. Implemented the Display Module

### **Mike**

1. Wrote the Render Tree portion of the Blog Post
2. Wrote the Progress Report
3. Implemented the Render Tree Module

## **Productivity**

We were very productive this sprint and accomplished our main goal of creating an end-to-end implementation which can take our web DSL and transform it into visible nodes in a dummy browser. However, we severely overestimated the difficulty of creating the render tree and being able to paint to canvas. As a result, we did not get to implementing tags

## **Grade**

While we did not manage to reach our excellent-scope goal of implementing tags, the difficulty of reaching an end-to-end dummy implementation was more challenging than we expected and we believe we still achieved Excellent scope when we factor in the amount of work we did. Constructing the render tree was especially challenging and we ran into many roadblocks, which often resulted in refactoring previously written code.

## **Goals**

### **Satisfactory**

1. Achieve full constraint styling capability with well-defined attributes as well as implementation of non-constraint-based styles

### **Good**

1. Be able to draw text on the canvas
2. Users will be able to use pre-implemented tags in the code to structure content

### **Excellent**

1. Users will be able to create custom tags
2. Users will be able to use functions and variables in the styling portion of the language to structure and style data