

Interactive Visualization Final Project Report

Recipe Visualization

Christopher Wong & Ross Macfarlane

Text-based recipes are abundant and lend themselves well to standard black-ink print, but with current technology it would be quite easy to create more vivid recipes that utilize modern visualizations. There are a couple of major issues that arise from the standard recipe formats found in most cookbooks. One issue is following the instructions; when a cook is running back and forth between the fridge and the frying pan, it is easy to lose track of which step is next in a large paragraph. The more time spent trying to understand a recipe, the less attention the cook can afford to give to the actual cooking process. By presenting the recipes in a novel way, we hope to overcome these issues that are common in cooking with recipes. Our goal was to develop this tool such that anyone could easily use our visualization while cooking.

With an event-based timeline, it is clear to follow the progression of steps and see their dependencies on each other. Additionally, it is possible to gauge the relative complexity of and attention required by a recipe. It could also be possible to determine which cooking appliances and tools are necessitated by a recipe based on the events.

For data collection and internal representation, we developed a set of functions that facilitates the process, if only by a small amount. Data collection was done manually, and hard-coded into the project files. This ended up being one of the most tedious tasks; if we had more time, we would have liked to developed a separate web tool to help users create recipes. As it currently stands, users would be unable to write recipes unless they modified the project files directly and became familiar with the functions we created and their nuances. Even if a tool was developed to help this, it would still be a time-consuming process, and not particularly worse or better than writing a recipe in traditional format. Many online communities exist where people share their recipes with the rest of the world; our tool could be another way to share recipes.

```

recipes.push({
  "title": "shrimp scampi with linguini",
  "author": "Tyler Florence",
  "servings": 4,
  "prepTime": 15,
  "cookTime": 25,
  "ingredients": {},
  "instructions": {},
  "results": {},
  "image": "/img/shrimpscampi.jpg"
});

recipes.select(2).addIngredient("linguini", "linguini", 1, 0, "lb", false)
  .addIngredient("butter", "butter", 4, 0, "tbsp", false)
  .addIngredient("olive oil", "extra-virgin olive oil", 4, 0, "tbsp", false)
  .addIngredient("shallots", "diced shallots", 2, 0, "", false)
  .addIngredient("garlic", "minced garlic", 2, 0, "cloves", false)
  .addIngredient("red pepper flakes", "red pepper flakes", 0, 0, "", true)
  .addIngredient("shrimp", "jumbo shrimp", 1, 0, "lb", 0)
  .addIngredient("salt", "kosher salt", 0, 0, "", false)
  .addIngredient("pepper", "ground black pepper", 0, 0, "", false)
  .addIngredient("wine", "dry white wine", .5, 0, "cup", false)
  .addIngredient("lemon juice", "juice of a lemon", 0, 0, "", false)
  .addIngredient("parsley", "finely chopped parsley leaves", .25, 0, "", true)
  .addInstruction("A1", "boil a large pot of water", [], [])
  .addInstruction("A2", "add 2 tbsp of salt and all of the linguini to the pot", ["salt", "linguini"], [prev])
  .addInstruction("A3", "stir to make sure pasta separates", ["linguini"], [prev])
  .addInstruction("A4", "cover pot", ["linguini"], [prev])
  .addWait(0, "allow water to return to a boil")
  .addInstruction("A5", "cook until slightly al dente", ["linguini"], [prev])
  .addWait(7, "until pasta is almost done")
  .addInstruction("A6", "remove from heat and drain pasta", ["linguini"], [prev])
  .addInstruction("B1", "melt 2 tbsp of butter with 2 tbsp of olive oil over medium-high heat", ["butter", "olive oil"], [])
  .addInstruction("B2", "saute shallots, garlic, and red pepper flakes", ["shallots", "garlic", "red pepper flakes"], [prev])
  .addWait(3.5, "until shallots are translucent")
  .addInstruction("C1", "season the shrimp with salt and pepper", ["shrimp", "salt", "pepper"], [])
  .addInstruction("C2", "add shrimp to pan and cook", ["shrimp"], [prev, "B2"])
  .addWait(2.5, "until shrimp have turn pink")
  .addInstruction("C3", "remove shrimp from pan and set aside; keep warm", ["shrimp"], [prev])
  .addInstruction("C4", "add wine and lemon juice to pan", ["wine", "lemon juice"], [prev])
  .addWait(0, "bring to a boil")
  .addInstruction("C5", "add remaining butter and oil (2 tbsp each)", ["butter", "olive oil"], [prev])
  .addWait(0, "when butter has melted")
  .addInstruction("C6", "return shrimp to the pan, along with parsley and cooked linguini", ["shrimp", "parsley", "linguini"], [prev, "A6"])
  .addInstruction("C7", "stir well and season with salt, pepper", ["salt", "pepper"], [prev])
  .addInstruction("OPT", "season with extra lemon juice and olive oil if desired", ["lemon juice", "olive oil"], [prev])
  .optional()
  .serve("serve immediately");

```

Above: A sample of code used to generate the recipe entry for “Shrimp Scampi with Linguini”.

This recipe is relatively simple compared to others, and is missing many optional properties that could possibly be used to improve the visualization. This format is only slightly better than writing out the object explicitly, and is fairly difficult to manage.

Our visualization is generated using standard HTML in combination with jQuery and D3.js. The ingredients and instructions list is generated from the recipe data using JavaScript functions that run at page load. The graph is generated using D3.js after some live preprocessing of the data. The color choice is determined by length of the timeline, simply using the distance

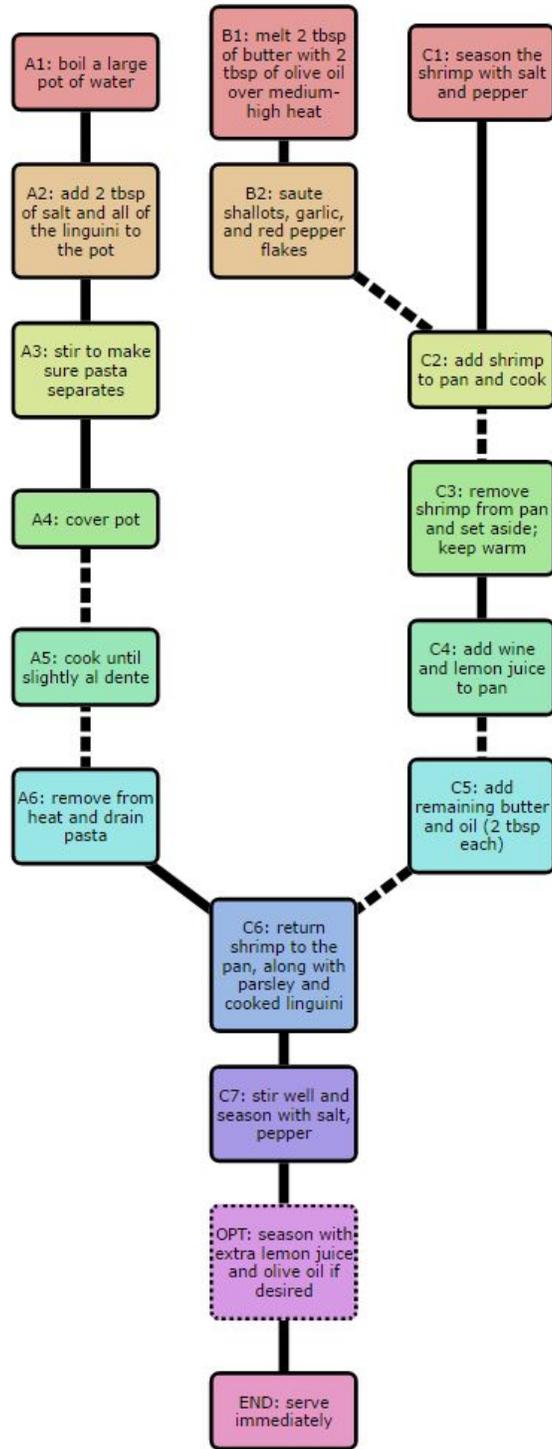
along the timeline to choose the hue value in the HSL color space. The algorithm that generates the layout of the graph is based on an algorithm developed previously by Ross that is currently used in professional software to visualize and help manage human tasks in large, team-based projects. This algorithm provides a level of abstraction between the node layout and the final rasterization that allows us to easily switch between left-to-right and top-to-bottom views. This previous work proves the algorithm's scalability with larger data sets of thousands of nodes with complex dependency structures (although we hope no recipe ever becomes this complex).

Amount	Ingredient
1 lb	linguini
4 tbsp	butter
4 tbsp	extra-virgin olive oil
2	diced shallots
2 cloves	minced garlic
-	red pepper flakes
1 lb	jumbo shrimp
-	kosher salt
-	ground black pepper
0.5 cup	dry white wine
-	juice of a lemon
0.25	finely chopped parsley leaves

- A1** boil a large pot of water
- A2** add 2 tbsp of salt and all of the linguini to the pot
  - ingredients: salt, linguini
  - dependencies: A1
- A3** stir to make sure pasta separates
  - ingredients: linguini
  - dependencies: A2
- A4** cover pot
- A5** cook until slightly al dente
- A6** remove from heat and drain pasta
- B1** melt 2 tbsp of butter with 2 tbsp of olive oil over medium-high heat
  - ingredients: butter, olive oil
- B2** saute shallots, garlic, and red pepper flakes

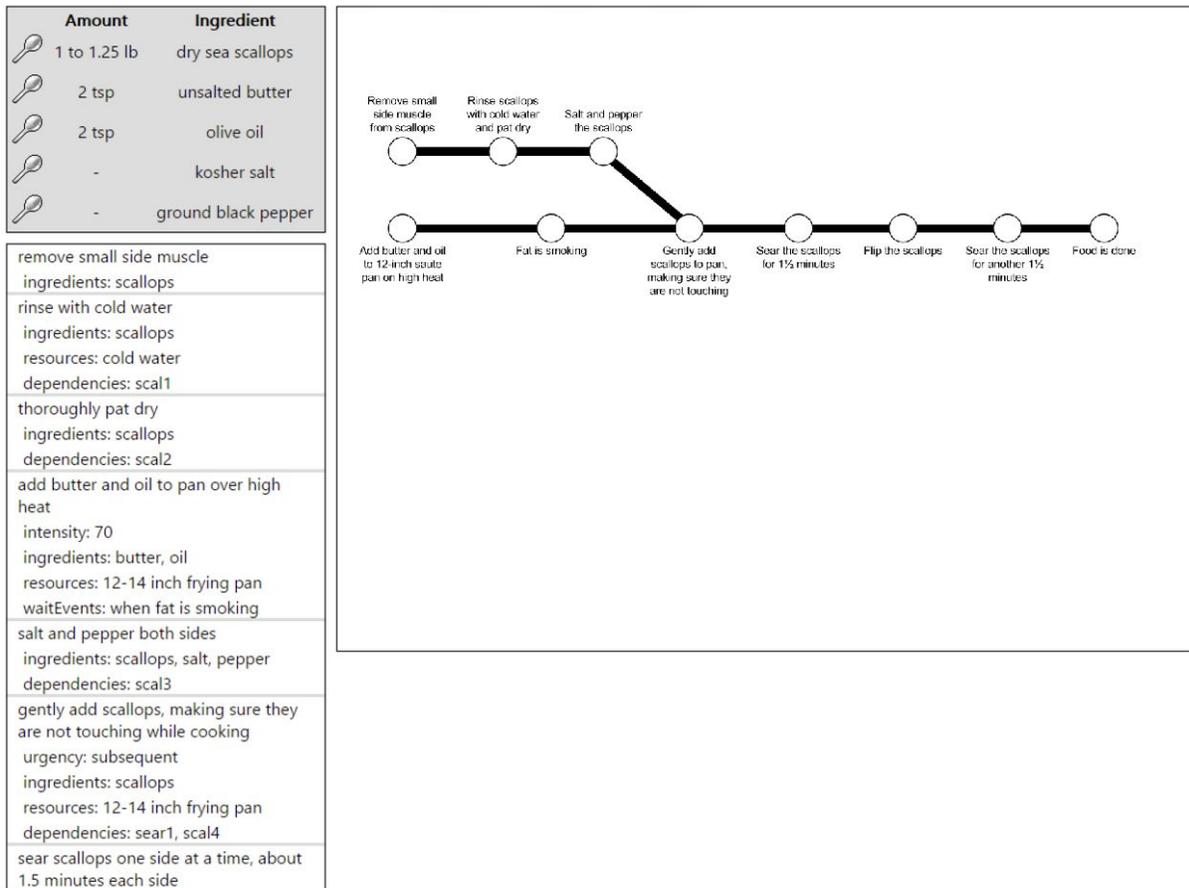
Left: The ingredients list. The background image is optionally supplied at creation.

Right: An excerpt from the instructions list for the same recipe. Individual instructions can be expanded to provide additional information about the step, provided at recipe creation.



Above: A complete timeline for this recipe. Steps can be toggled on or off to denote which tasks have been completed. The dashed lines indicate dependencies with associated waiting events.

In our initial design for this visualization, we wanted to have icons associated with each ingredient to help indicate how the ingredients tie into the instructions. We eventually decided to scrap this idea because it was not feasible to create or find unique icons for all the ingredients and they ended up making the instructions list too busy. Most of the other core aspects of our visualization did not deviate very far from our initial plan, although a number of features are currently incomplete, and have partial code written for them that is not used in our final product.



Above: An early prototype of our design. The graph used here was a static image create manually in Google Drawings.

In our “pilot study” that we held in class, we asked classmates for general feedback on our visualization, demonstrating the features with the above prototype. Overall we received positive feedback on the underlying concept. We specifically asked classmates what features they felt were missing from our visualization and they felt as though some of the “collapsed” hidden information in the instructions list had to be shown by default, and that relevant ingredients should be shown alongside instructions in the timeline. Some other recommendations included: support for smartphones and tablets; “printable” versions separate from the default view; using edges to display additional information such as wait times or optional dependencies; a toggle for topdown or leftright views of the timeline; streamlined instruction views where only the active instruction is expanded; and sequential step-by-step mode where the user would simply press “next” to advance to the next instruction.

This feedback lead us to implement the ability to toggle between top to bottom and left to right options for the graph, although this setting is not configurable on the page itself and must be set beforehand. We also implemented the suggestion for showing wait times and events on the edges by a combination of dashed lines and tooltips indicating what is being waited on. While we were not able to fully implement a sequential step-by-step mode, we added a feature that allows users to “check off” tasks as completed. Features that were suggested that we were unable to implement in time included variable edge length based on cooking time, and mobile support.

Were we to have infinite time and budget to conduct a user study, we would likely advertise and offer free cooking sessions at cookingware stores with kitchens and compare the traditional recipe format to our format. We would survey users asking how easy it was to follow the same recipe in each format and compare the results. This way we could reach out to a

relevant audience and have the study take place in a controlled environment and choose which recipe is used. We would likely include questions with responses on the Likert scale asking if they prefer our visualization over traditional recipes in paragraph format and ask for additional optional feedback to see how the visualization could be improved.

Our visualization features a number of interactive components that significantly improve its usability. When the user hovers their mouse over a node in the graph or an entry in instructions list it will be highlighted along with the corresponding node or instruction and the ingredients required by this step. The user can also click on instructions to toggle the additional properties list for its entry in the instructions list. By clicking on the instruction ID on the list, a step can be marked as completed, which will increase its transparency along with its corresponding node in the graph. The dashed lines indicating the dependencies with a wait event can be hovered over to show a tooltip that explains what is being waited on.

The work on this project was pretty evenly split between the two of us, as all code and stylesheets were written using the collaborative IDE Cloud9, and we typically developed features together simultaneously. Features that were not collaboratively developed include the data collection and internal representation, which was mostly Christopher, and the tree-construction algorithm and rendering of the D3 graph, which was mostly Ross.

Recipe Sources:

Florence, Tyler. "Shrimp Scampi with Linguini." *Shrimp Scampi with Linguini Recipe*. Food Network, n.d. Web. 05 May 2016.

Brown, Alton. "Seared Scallops." *Seared Scallops Recipe*. Food Network, n.d. Web. 05 May 2016.