

Undergraduate Research Final Report

Data Visualization and Representation of a
Quantitative Patient State in Radiation Oncology

Liza Mathews

Mentors: Dr. Todd McNutt, Dr. Russell Taylor, Michael Bowers

Abstract

This main purpose of this project was to improve upon certain features of charting tools within OncoBrowser. I specifically focused on the star chart (radar chart) that helps clinicians visualize the patient condition. I used d3.js (a JavaScript charting library) to make a new star chart that was faster and more interactive (mouse over tooltips, highlighting relevant data). Additionally, I built a new user interface using JavaScript and Bootstrap and connected it to a SQL backend (for testing purposes). In order to connect the frontend and backend, I used Knex.js (a SQL query builder for MySQL) and Express.js (a web framework for Node.js). Instead of adding these features directly to OncoBrowser, I will link to the web page I created. In the future, I plan to do this with longitudinal patient outcome charts, as well as other charting tools that need new, more interactive features.

Significance and Motivation

The goal of this project was to make charting tools easier for clinicians to use and modify data with. By adding more interactive features to existing charting tools, clinicians will be able to visualize the patient condition in a more efficient and clear manner.

Project Accomplishment Overview

The minimum deliverable of this project was making the star chart more interactive. This was accomplished by adding mouse over tooltips, a feature that highlights relevant data, and buttons that allowed clinicians to modify (add/delete) data points. The expected deliverable of this project was to make the star chart more interactive and have proper code documentation.

The code is available on GitHub and has appropriate commenting. The maximum deliverable (which was not accomplished) was to have an interactive star chart and a longitudinal patient outcome chart. Given the time constraints, this was not possible. There were unforeseen technical difficulties with setting up the remote desktop, which cut down on the amount of time I was able to spend on this project.

Future Work Proposal

In the future, I plan to make a new longitudinal patient outcome chart module, as well as improve other charting tools that need new, more interactive features.

Background Overview

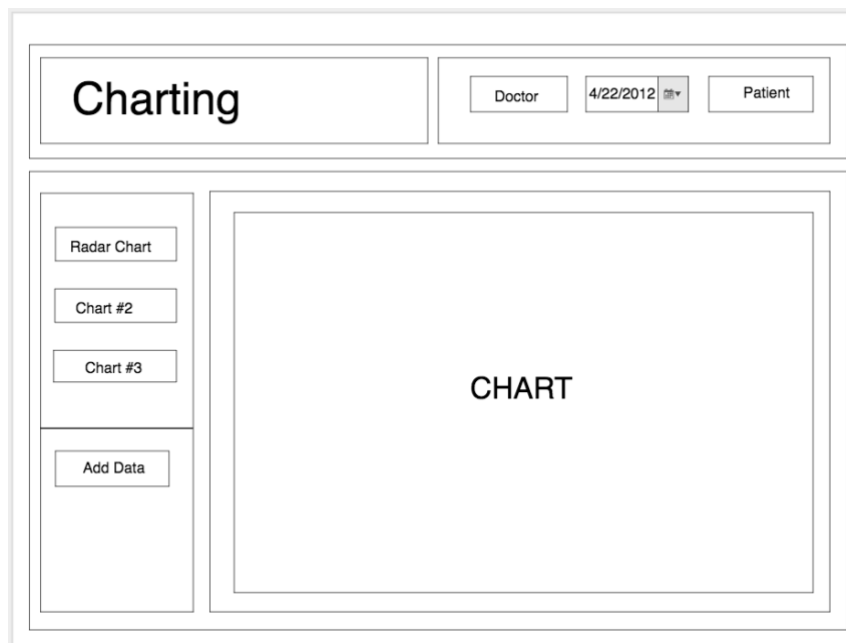
The existing charting system is written with C#/ASP.NET and uses MS Charts. There are not many charting tools in these languages that offer interactive features; however, I researched many potential JavaScript charting libraries that did (Plotly.js, Charts.js, and d3.js). Overall, d3.js offered the most flexibility and had the newest features. As far as integrating charts written in JavaScript into a frontend that was written in C#/ASP.NET, this was much more complicated. It is possible to embed HTML into ASP.NET without many issues. However, connecting data to the charts requires an AJAX call, meaning that the JavaScript communicates directly with the server through the JS XML HTTP Request object. Also, connecting JavaScript with a SQL backend requires Knex.js and Express.js. Knex.js is a SQL query builder that is designed to be flexible, featuring node style callbacks, a promise interface, and full featured query and schema

builders. Express.js is a web application framework for Node.js. Finally, Bootstrap offers a flexible HTML/CS/Javascript library that helps user interfaces look more presentable.

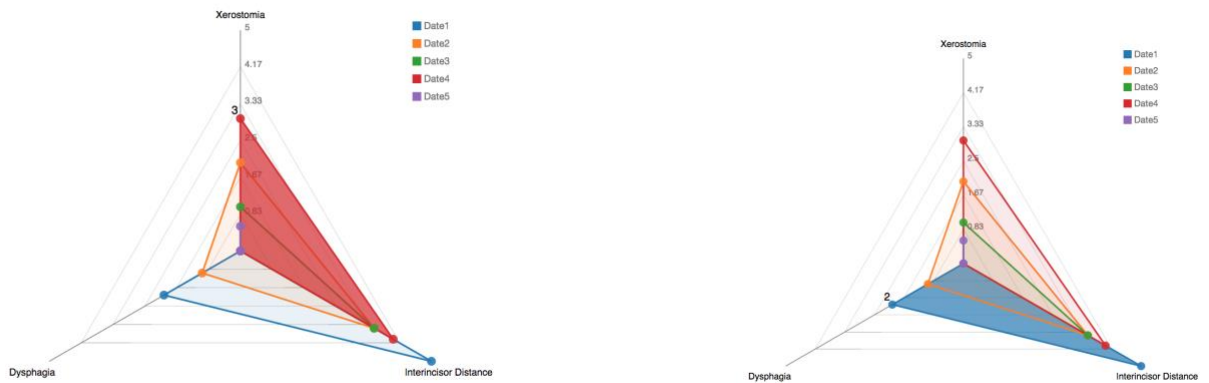
Proposed Solution/ Approach

My proposed solution was to integrate the JavaScript charting tools into the existing framework. However, after running into numerous difficulties, I decided to build a completely new front end with the JavaScript charting tools (and provide a link to these charting tools on the OncoBrowser site). I planned to write the new interactive radar chart using the d3.js charting library. The proposed schema for the backend data was to have a column for a patient ID, the assessment, the grade, and the date the assessment was taken.

The following is a wireframe diagram of the proposed webpage.



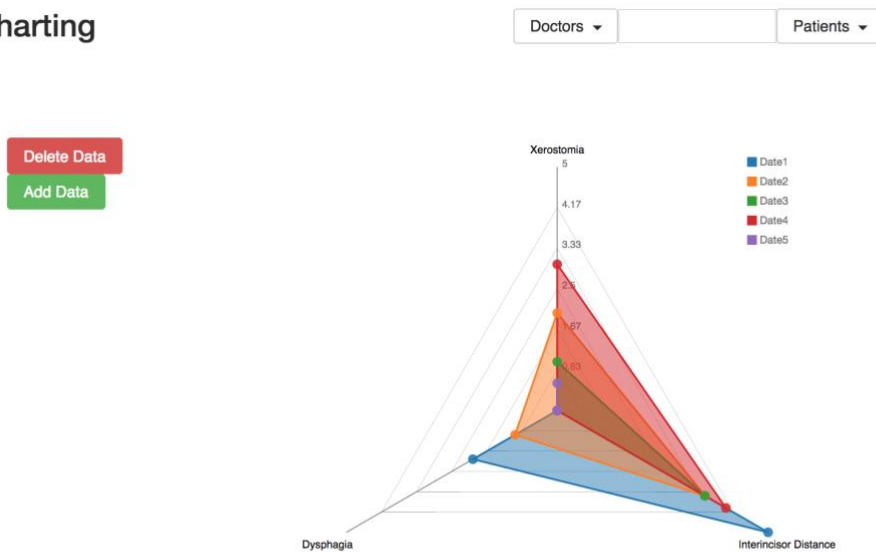
Results



These images show the highlighting and tooltip feature in the new JavaScript star chart.

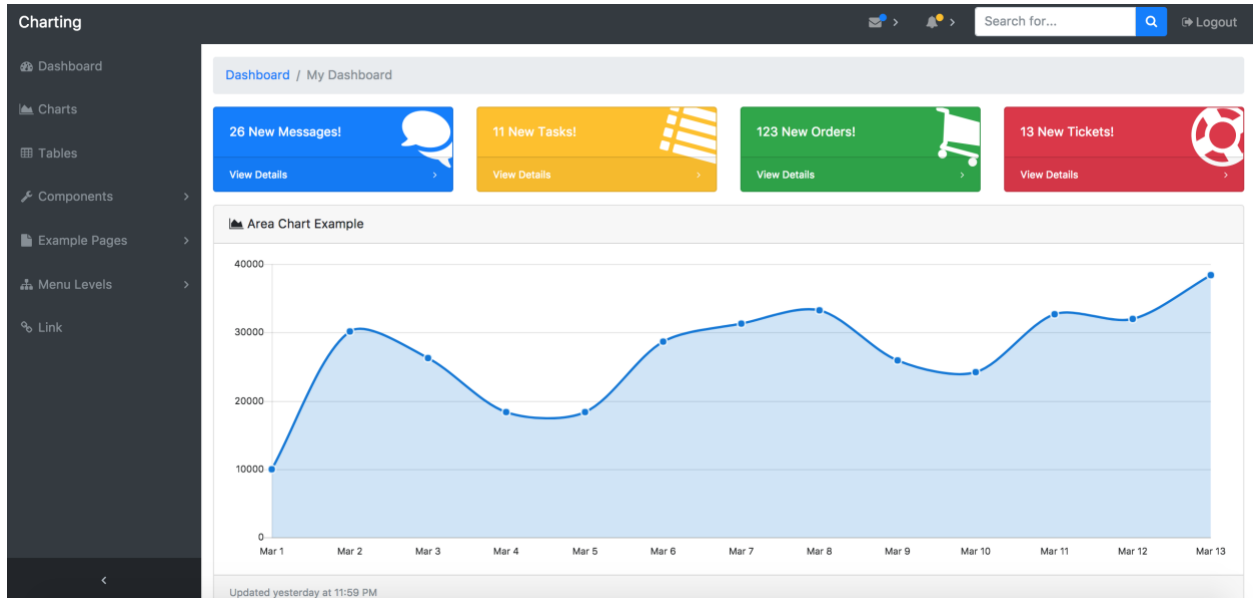
The following image is the first web page I built (without Bootstrap theme and backend). It was modeled after the wireframe diagram shown in the previous section.

Charting

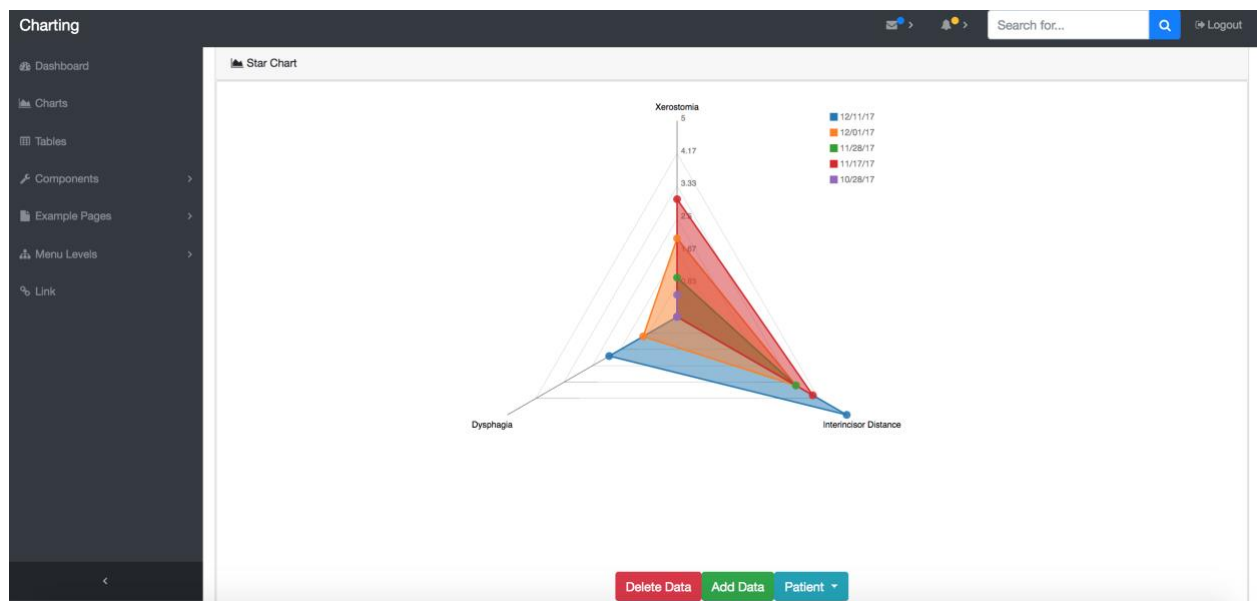


After adding the bootstrap theme, I added a home page and a side bar that featured other components (for possible future work)

This image shows the home page view.



This is the Star Chart Module (under the charts section of the side bar). It features three buttons that allows users to add/delete data and toggle the patient data shown in the graph.



Using SequelPro and Knex migrations, I made a sample database to show how the queries would work once the frontend was connected to OncoBrowser's backend.

(MySQL 5.7.21) Local/dummy_charting_node_database/assessment

Select Database Structure Content Relations Triggers Table Info Query Table History Users Console

Search: patientID = Search Filter

patientID	toxicity	grade	date	created_at	updated_at
342	Dysphagia	2	2017-12-11	2018-01-22 15:20:02	2018-01-22 15:20:02
342	Dysphagia	1	2017-12-01	2018-01-22 15:22:02	2018-01-22 15:22:02
342	Interincisor	5	2017-11-28	2018-01-22 15:22:16	2018-01-22 15:22:16
342	Dysphagia	1	2017-11-20	2018-01-22 15:22:40	2018-01-22 15:22:40
340	Dysphagia	1	2017-12-10	2018-01-22 15:23:04	2018-01-22 15:23:04
340	Dysphagia	2	2017-12-02	2018-01-22 15:23:21	2018-01-22 15:23:21
340	Dysphagia	1	2017-11-27	2018-01-22 15:23:46	2018-01-22 15:23:46
342	Interincisor	4	2017-11-20	2018-01-22 15:24:01	2018-01-22 15:24:01
339	Dysphagia	1	2017-12-11	2018-01-22 15:24:24	2018-01-22 15:24:24
339	Dysphagia	2	2017-12-01	2018-01-22 15:24:39	2018-01-22 15:24:39
339	Dysphagia	2	2017-11-28	2018-01-22 15:25:09	2018-01-22 15:25:09
339	Dysphagia	1	2017-11-20	2018-01-22 15:25:24	2018-01-22 15:25:24
319	Dysphagia	3	2017-12-11	2018-01-22 15:27:16	2018-01-22 15:27:16
319	Dysphagia	1	2017-12-01	2018-01-22 15:27:27	2018-01-22 15:27:27
319	Dysphagia	2	2017-11-28	2018-01-22 15:27:43	2018-01-22 15:27:43
342	Interincisor	4	2017-11-20	2018-01-22 15:28:02	2018-01-22 15:28:02
342	Xerostomia	3	2017-12-11	2018-01-22 15:28:37	2018-01-22 15:28:37
342	Xerostomia	2	2017-12-01	2018-01-22 15:28:39	2018-01-22 15:28:39
342	Xerostomia	2	2017-11-28	2018-01-22 15:28:41	2018-01-22 15:28:41
342	Xerostomia	1	2017-11-20	2018-01-22 15:28:44	2018-01-22 15:28:44
340	Xerostomia	2	2017-12-11	2018-01-22 15:28:48	2018-01-22 15:28:48
340	Xerostomia	3	2017-12-01	2018-01-22 15:28:50	2018-01-22 15:28:50
340	Xerostomia	3	2017-11-28	2018-01-22 15:28:52	2018-01-22 15:28:52
340	Xerostomia	4	2017-11-20	2018-01-22 15:28:54	2018-01-22 15:28:54
339	Xerostomia	1	2017-12-11	2018-01-22 15:28:55	2018-01-22 15:28:55
339	Xerostomia	2	2017-12-01	2018-01-22 15:29:03	2018-01-22 15:29:03
339	Xerostomia	3	2017-11-28	2018-01-22 15:29:06	2018-01-22 15:29:06
339	Xerostomia	4	2017-11-20	2018-01-22 15:29:10	2018-01-22 15:29:10
319	Xerostomia	1	2017-12-11	2018-01-22 15:29:13	2018-01-22 15:29:13
319	Xerostomia	2	2017-12-01	2018-01-22 15:29:17	2018-01-22 15:29:17
319	Xerostomia	2	2017-11-28	2018-01-22 15:29:20	2018-01-22 15:29:20

TABLE INFORMATION

- created: 1/22/18
- engine: InnoDB
- rows: 51
- size: 16.0 KiB
- encoding: utf8

51 rows in table

Appendix

<http://knexjs.org/>

<https://expressjs.com/>

<https://d3js.org/>

<https://getbootstrap.com/>