CS5500 - Foundations of Software Engineering
Northeastern University, Summer 2021
Prof. A. Slaughter

Group Project 8
Stephanie Chung, Ricardo Garay, Luke Parkhurst, Kayla Sear
Final Sprint
8/19/2021

Github Link: https://github.com/stephechung/tracey
Trello Link: https://trello.com/b/jagzLmFL/cs5500-final-project

Delivered Features:

In the last sprint, we worked on building out our API controllers and connecting html files to display a variety of web pages for the user. We built three different view pages for the Tracey application that shows an overview of both the location and activities collection. In addition, we made a home page that the user can use to navigate the different pages.

As a web user, I want to be able to see a page of locations I've visited to know which places I've been to the most.	With bootstrap, we were able to build a web page that displays a table of locations that has names associated with it including id, latitude, and longitude. Endpoint: ***localhost:8093/location***
As a web user, I want to see a web page of the activities that I've done so I can keep track of metrics such as total calories that I've expended.	We created another view page of the last 10 activities from the activities collection in a table format with a summary of total metrics such as sum, median, and average for steps, duration, distance and calories. Endpoint: ***localhost:8093/activity***
As a web user, I want to be able to see the total number of calories burned per day/week/month over time	We used a third library, highchart.js for the third view page that displays a bar chart that shows how many calories for each activity type the user has burned over a given period of time. Endpoint: ***localhost:8093/activity2***

View 1: List of Locations

Locations

ID	Name	Latitude	Longitude
6552482	Home	47.67645	-122.32305
7187779	Forza Coffee Co.	47.678776	-122.32676
7256266	Fred Meyer	47.723316	-122.29199
7772456	Chuck E. Cheese's	47.630173	-122.14488
8776300	Green Lake Elementary	47.676006	-122.328705
8291699	A.R.O.	47.597862	-122.32803
8291698	School	47.67656	-122.32843
12766926	Blick Art Materials	47.6155	-122.32066
12803263	ZoÃ«yogurt	47.678867	-122.32608
16182633	Mona's Bistro	47.675724	-122.32562
18505034	Whole Foods Market	47.674988	-122.316635
19360667	QFC	47.662308	-122.2968
19560063	Garage Billiards	47.612427	-122.320435
20364064	Green Lake United Methodist Church	47.675663	-122.328255
21631794	Fuji Sushi	47.600124	-122.32665
22397993	Honey Court Seafood	47.597748	-122.32489
22540733	Biscuit Bitch	47.610466	-122.34175
22540735	The Pink Door	47.610302	-122.34246
24802706	The Home Depot	47.71405	-122.34338
25230592	Jo-Ann Fabric and Craft Stores	47.740036	-122.343925

View 2: Activity Metrics

Activity Metrics

Querying the database for all activities and finding the sum, average and median metrics among other things

Metric	Duration	Steps	Distance	Calories
Sums	4047738.0	5594926.0	3610264.0	200866.0
Averages	272.5566	376.73734	243.09904	13.525419
Medians	154.0	230.0	143.0	7.0

Last 10 Activities

Name	Duration	Steps	Distance	Calories
walking	22	44	22	1
walking	36	72	36	2
walking	69	138	69	3
walking	9	18	9	0
walking	154	234	154	8
walking	76	152	76	4
walking	412	550	412	20
walking	62	123	62	3
walking	395	572	395	19
walking	35	44	35	2

View 3: Chart view of total calories burned per month

CodeMR Report:

An update to the CodeMR report reveals that not much has changed from Sprint 2. There has been an overall improvement to the cohesion and coupling. But after Sprint 2, not much change was further created. The major area of concern still resides in the SwaggerGeneratedCode package where the main bulk of cohesion issues lie. The rest of the code base, other than the new Swagger Generated Code, from Sprint 1 into Sprint 2 showed incredible improvement, eliminating all cohesion and coupling issues to under 5%.

This improvement was expected as we built the code base with Object Oriented Programming in mind. Throughout the project, as the Spring API and Restful API was inserted, a mindset of keeping coupling and cohesion low was created as a standard. While some concerns arise from the number of external imported classes, this cannot be avoided due to the importation of required API's to create the basic web application.

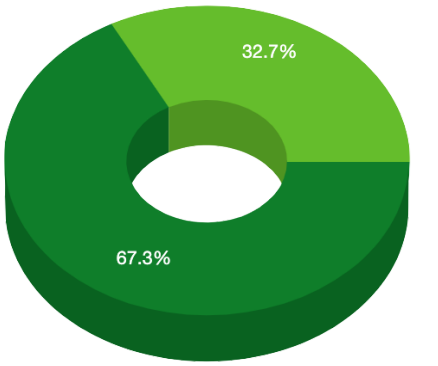
As seen in figure 1, coupling and complexity was kept low as average. This can be compared in figure 2, which is from Sprint 2. The only area of concern is from cohesion, though it maintains at the same rate as Sprint 2, but is vastly improved from Sprint 1, as seen in figure 3. Lack of cohesion breakdown in figure 4 shows that the majority of issues come from the Swagger generated code. Figure 5 shows a comparative to figure 4 where the majority of lack of cohesion was originally coming from the model package. A final comparison can be made from Sprint 3 to Sprint 1 in figures 6 and 7 respectively. The

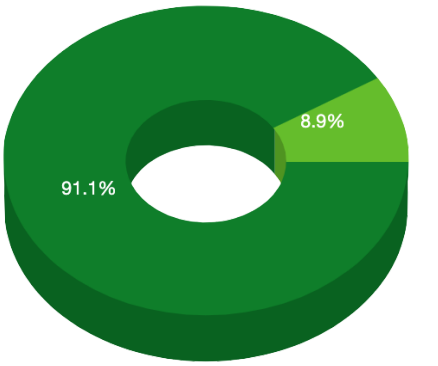
coupling stayed low throughout the entire project, even when the number of packages doubled.

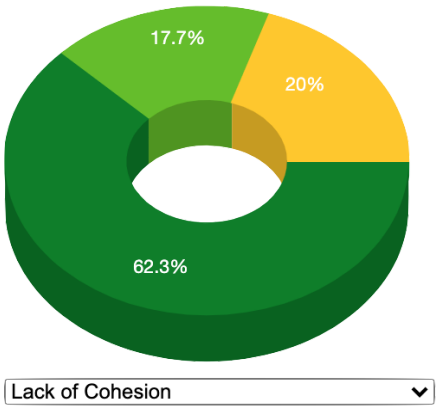
Overall, these figures show a tremendous improvement in code stability from Sprint 1 into Sprint 2. This stability can then be seen as being maintained into Sprint 3, even after importation of other code bases.

Below are CodeMR reports from the most recent (Sprint 3) to the first (Sprint 1) presentations as evidence of improvement in the different areas CodeMR tracks. Also attached, Cohesion Breakdown in Treemap view and Coupling and Cohesion Reports in Package view.

Figure 1: Sprint 3 Metrics


32.7%
67.3%
Complexity


8.9%
91.1%
Coupling


17.7%
20%
62.3%
Lack of Cohesion

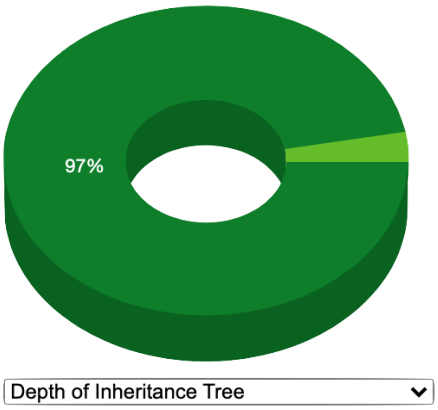

97%
Depth of Inheritance Tree

Figure 2: Sprint 2 Code Metrics

Figure 3: Sprint 1 Code Metrics

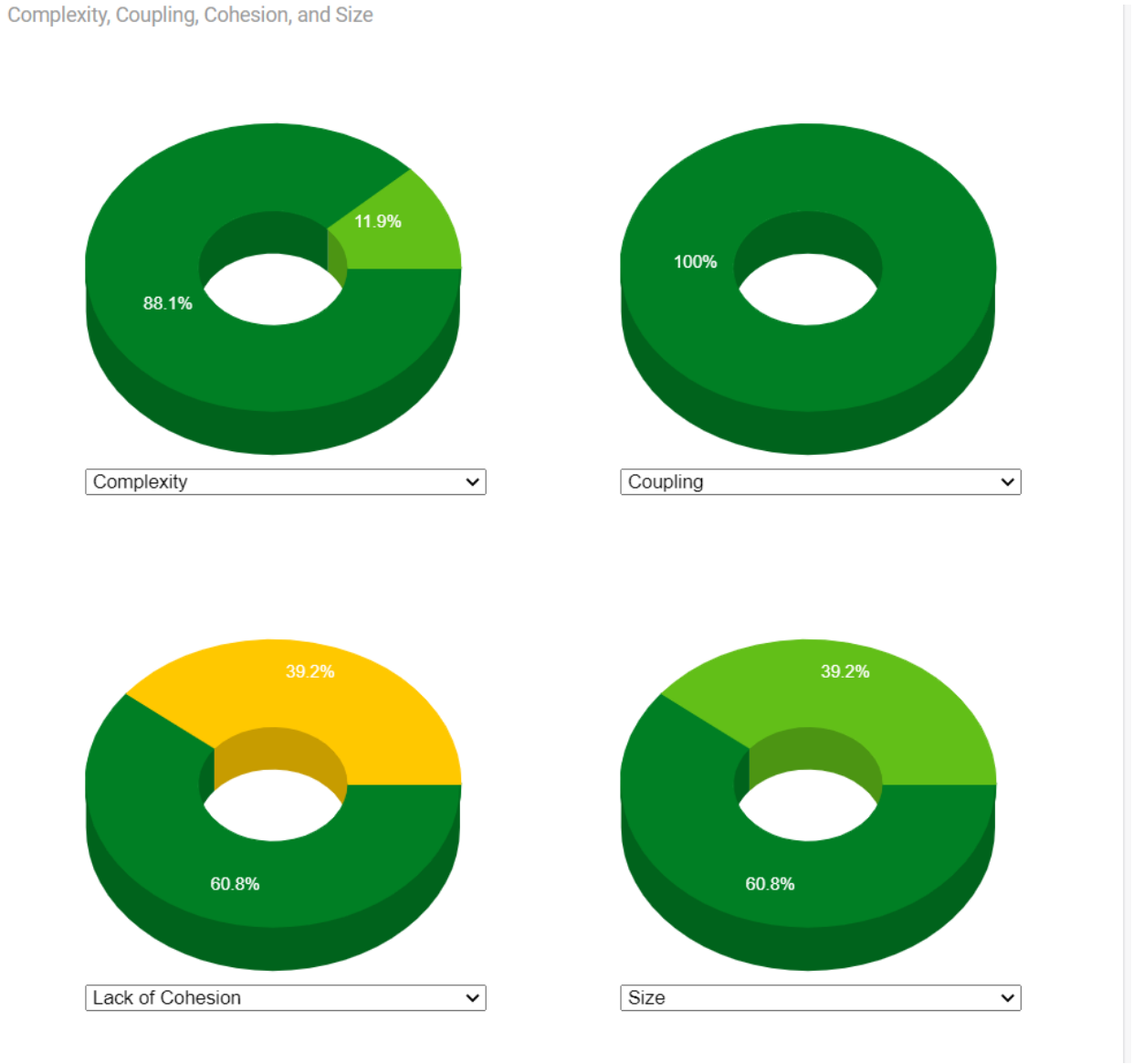


Figure 4: Sprint 3 Cohesion Breakdown Report

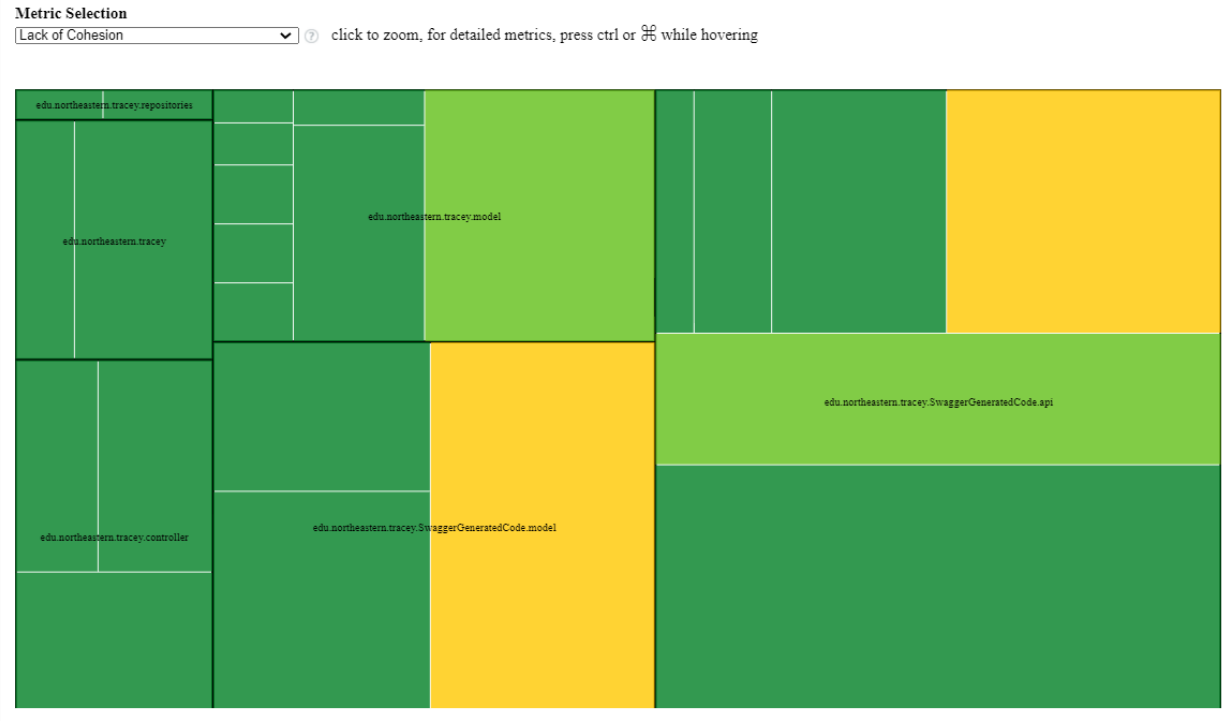


Figure 5: Sprint 1 Cohesion Breakdown Report

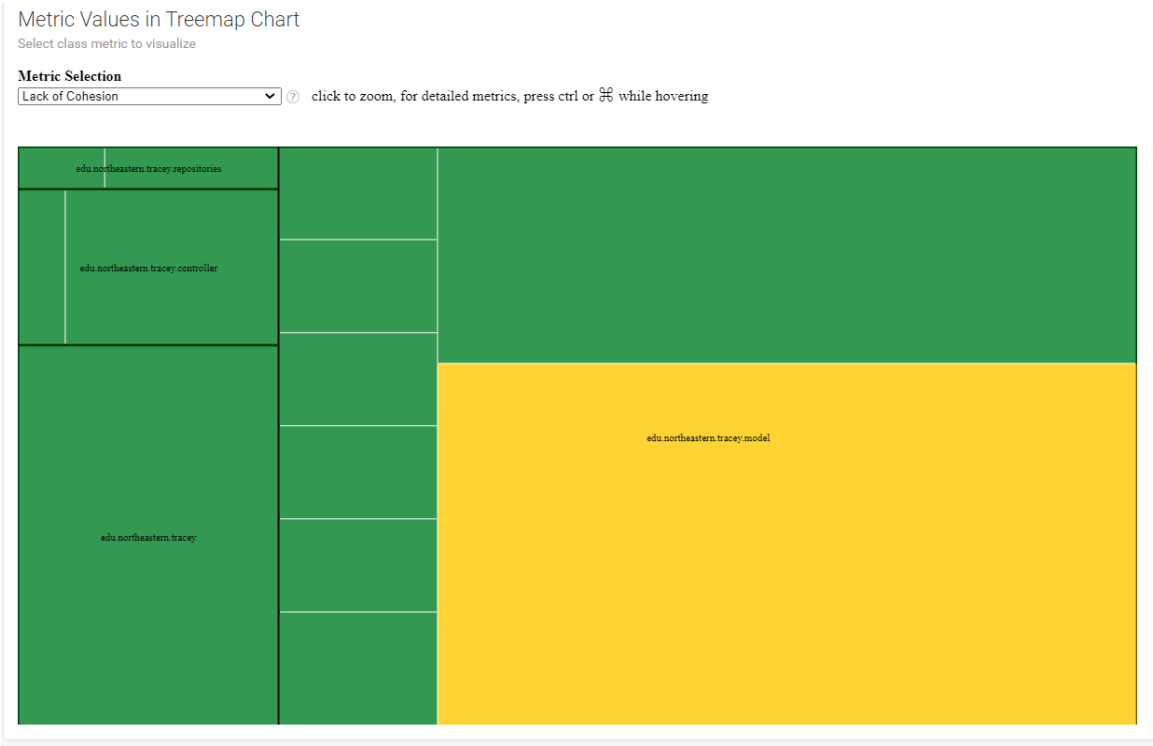


Figure 6: Sprint 3 Coupling Report

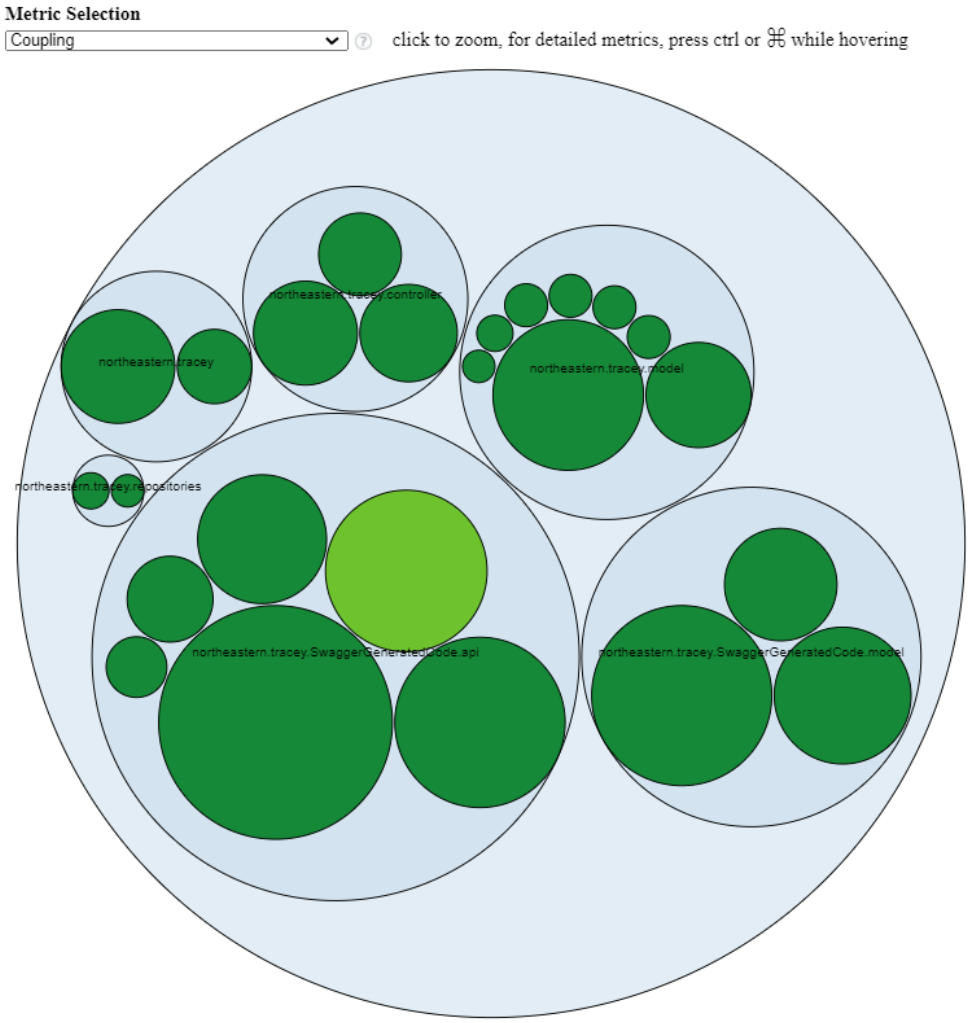


Figure 7: Sprint 1 Cohesion Report