Campus Catalog

Phase 4 Technical Report

Andrew Luo(Phase Leader), Brandon Hinh, Richa Gadre, Ryan Gahagan, David He

Dec 6, 2021

I. Motivation

University is not just a place to learn. It is a place where you will find a community filled with new people and experiences. It is a place where you will make friends for life and explore your own potential. It is a place where you will discover new opportunities and reach new heights. It is a place where you start a new chapter of life. However, a question remains: how do you decide which university to attend?

We created campuscatalog.me to help people make more educated university decisions. 43.2 million students in the U.S are in debt by an average of $39,351 each and we would like to help. By providing a wide range of helpful information, our website will help students make one of the most important decisions in their life. We want to help you find the best university option for the price you deem acceptable by providing information regarding the university, cost of living on or off-campus, and other amenities available.

II. User Stories

Here's all of the user stories we gave, with responses in blue:

Course Prerequisites

"When I was looking at the page for a specific course, I was not sure if I would have to take another class before being able to take this one. That is a little inconvenient and gets in the way of planning a schedule. I think it would be helpful to list the courses that must be taken before being able to take a specific class. Would that be possible to Implement?"

Hi Andrew! Thank you for your feedback. We definitely agree that listing the prerequisites for the class would be an amazing feature to include. Unfortunately, our api's do not provide that information and due to timing constraints we were not able to manually scrape a website, such as the UT course schedule. If time permits, we would

love to look into implementing this feature in the next phase.

Filtered Class Comments

“Hello developers, I've used rate my professor for my course searches before and there were many misleading comments on there. There were many comments that people have downvoted and people find them unhelpful. Would it be possible for you to display only useful comments?”

Hello Huan! We agree that comments sections should be optimized so that more useful comments are displayed towards the top. As of now, we have still not been able to implement a commenting feature, but will keep in mind to add a ratings feature for the comments and be sure to sort by the cumulative rating.

Class Times Per Professor

Hello, I'm currently a student looking at various courses that are offered but I've noticed that many of the courses don't have a professor listed. There's a specific professor I want to take for one class with no professors listed, and a friend of mine told me that I can use the times that professor usually teaches as a good way to predict which of these listings will be taught by the professor I want. Can you display the times a professor has taught a certain class in recent years, or alternatively provide multiple past syllabuses to help me predict this?

Hello Ryan! Thank you for your feedback! We agree that more information should be available concerning the course in case of a professor not being listed. Unfortunately,the apis we use only have the information for current semester report and the current syllabi registered for the courses of the semester. We doubt we will be able to get information that spans further back in time, but we will keep this in mind for any future updates to the website.

Polish department images

“As a user, the images for the departments are very unappealing. I would like clearer display of the images for each department. It seems like a image resolution issue.”

Hello Brandon! Thank you for your feedback! We definitely agree that the clarity of the images greatly hinders the aesthetic of the page. We are quite confused as to why we are getting such low quality resolution despite using an API like google images. We will

probably not be able to improve on this feature for the current phase, but we plan to polish this by phase IV.

Sorting Departments by Number of Research Labs

"I am a student interested in participating in research. If I do switch labs, I would like it to be in the same department. Therefore, I am more interested in departments with more research labs. If I was able to sort based on number of research labs, I would be much more well equipped to pick a department to do research."

Hi Richa! Thank you for your feedback! We understand how important research opportunities are to students in deciding which classes/majors to take. It is unfortunate that despite the many research labs that are in UT, we do not have a central data source for this information. We would be definitely open to implementing this feature in the future if we found a source for this information.

More information on class attendance policy

> *"Hello, I am a user worried about the risks of disease transmission during classes. I would like an option to filter classes by their attendance policy, whether in-person, fully online, or a hybrid of both. This will help students feel more comfortable with the classes on their schedule."*

Hmm it might be a little bit difficult to sort based off of Corona restrictions since that information isn't available outside of UT's credential system (aka outside the course schedule). However, we are going to be pulling syllabus information through web scraping eventually, and we can incorporate their coronavirus policy within the sorting process most likely. This user story is more dynamic so we will tackle it in the future phases!

See which courses/professors are more project based

> *"I'm a user who's thinking about which classes I want to take next semester. I personally find that I learn a lot better in classes where the grades are focused on projects rather than tests, but I also have friends who prefer testing and homework to projects. Is it possible to provide me a way to view only classes that are more project based or more test based?"*

Unfortunately, we most likely will not be able to implement this feature, since we do not have an API to pull course specific information about projects, tests, etc. Additionally, we don't know if we can scrape the information either, since most classes don't seem to provide an up-to-date course syllabus. We will still, however, try to see if this is a viable option while scraping for course information in the future.

Splash page summary

> *"As a user, I want to be able to have some sort of idea about what the website is about from just the splash page. Maybe by adding a brief summary or the purpose (one or two sentences) of the website somewhere on the splash page. There could maybe also be some descriptive or eye-catching media on the page as well."*

I wasn't able to find a good way to add sentences to the splash page without it feeling a bit cluttered, but I added some additional info about the purpose of the website in the About Us page. To fill up the empty space on the splash page, I made the search bar (which isn't functional yet) a little larger, and added some large buttons that allow users to jump directly to the model pages. It definitely looks a lot better now with those changes.

Which courses have a smaller workload

> *"Hi development team, I am a student who is trying to balance out job hunt and school work at the same time. I would like to find courses with a smaller workload so that I can spend more time on job hunting and other activities. Is there a filter that I can use for finding courses with a smaller workload?"*

This information should be present in our website! There are ratings which specify the difficulty and quality of each course. In phase 2, we'll be adding sorting in which we'll sort each course by their rating, so you can sort and find the classes that are super light weight and easy and breeze right through college!

Extra Materials

> *"I am a user who wants to save money! I would like to see which courses require extra materials which will cost me money (such as textbooks, iclicker, calculator, etc), so I can pick a cheap class. Please make a sortable attribute for how much extra money I will have to pay for this class."*

This user story is a little bit outside of the scope of what we're aiming to do, but I can definitely see the benefit of it. If the materials are listed on the class websites, we can search them on maybe an Amazon API and determine the base process for a particular textbook or iClicker or Mathematica, but it would be much more difficult if that can't be found on Amazon or has student discounts attached to it. However, when working with the coronavirus user story, we can do a similar mechanism to sort by the budget in a later phase.

More info regarding class size

> *"Hi, I am a student who enjoys more interactions with the professor. I find it extremely hard to do in a class with over 50 students. Would it be possible to see how many students are usually enrolled in each course?"*

We were able to scrape the size of each section and our website will show the maximum number of students in each course. In Phase 3, we will add the ability to sort/filter by this value so you can quickly find exactly what you need.

Get professor ratings for specific courses

> *"Some professors are better at teaching some subjects than other subjects. As a user, I would appreciate the ability to see what other students rated a class taught by a specific professor. This would come in handy when choosing classes, particularly when multiple professors teach multiple classes."*

We are planning to add a ratings section hopefully in the next phase by implementing a comments and numbered ratings feature for users to fill out. This will allow us to aggregate all the responses and display the results to see.

See if Professors allow auditing for each course

> *"Hello, I'm a college student with interest in many different classes across several majors. I'd love to take more classes, but having too many grades makes life extremely stressful; so, I often like to audit (i.e. attend a course without any grade or credit) a course that sounds interesting but isn't relevant to my major. Is it possible for me to check whether a Professor or class typically allows students to audit that course?"*

Auditing is a common practice, and I'm sure it would be really helpful to have that information on our website. Unfortunately, we were unable to find a centralized location which contains this information. If you know of one, please let us know and we can attempt to scrape it in the next phase of the project.

Course grade distribution

> *"Course grade distributions could be a good indicator of how a course will be like. As a user I would like to be able to see previous grade distributions for a course. This could be in a form of a table/chart or a graph."*

This data is obtainable by scraping this page - https://reports.utexas.edu/spotlight-data/ut-course-grade-distributions. Unfortunately, we did not have time to implement it for this phase, since the website provides the data in one large image. We will definitely make an effort to scrape it in the next phase of the assignment since the data is really useful.

Professor Activities

> *"Hi, I am a student who likes to get involved on campus. I would like to see if the professor is involved in research or organizations around campus. For example, I would like to know if the professor chaperones study abroad programs or works in a lab with students outside of teaching."*

We know it's important to get background information about professors. Any professor with a CV on the UT syllabus website will have their CV displayed on our website. Their instance page will also list their publications, which should give you some idea about their research interests. This may not necessarily include all the information you wanted, but hopefully it will give you a starting point to look further into a professor's involvements.

Here's all of the user stories we received, along with our responses:

User Story: Housing comparison tool

> *"Hello, I'm a student currently looking for housing near my university for the upcoming school year. As of now, when using your website, I can only look at one housing option at a time which makes it harder to compare different different housing options at the same time without changing what page I'm on. Would it be possible to add some sort of page/tool to the website in order to view the details and attributes of two different housing options of the user's choosing side-by-side on a single page?"*

Hi Ruben, I don't think we will be able to implement that type of feature for our project due to time constraints. However, it does sound like an very interesting feature to add, we will take your opinion into consideration during phase 4.

User Story: Explanation of Transit and Walk Scores

Hello, I'm a user interested in learning more about the benefits and rankings of different housing locations. However, I'm not sure what the transit and walk scores mean. Could you add a paragraph explaining what the scores are and maybe what factors go into calculating the scores?

Hello @victoriahall,

During this phase, we will implement a filter for our housing page which categorizes the score into 5 divisions such as walker's paradise, somewhat walkable, and car-dependent. We believe these terms can be easily comprehended so hopefully, they address your needs.

User Story: Housing comparison tool

Hello, I'm a student currently looking for housing near my university for the upcoming school year. As of now, when using your website, I can only look at one housing option at a time which makes it harder to compare different different housing options at the same time without changing what page I'm on. Would it be possible to add some sort of page/tool to the website in order to view the details and attributes of two different housing options of the user's choosing side-by-side on a single page?

Hi Ruben,

I don't think we will be able to implement that type of feature for our project due to time constraints. However, it does sound like an very interesting feature to add, we will take your opinion into consideration during phase 4.

User Story: University Events

Hi developers, I am a user that loves participating in school events since I have a lot of school spirit. Some of these events include university football games, Halloween events, and other university affiliated social events. Could these be included in the Events page?

Hi Varad,

Although your idea sounds interesting, it doesn't align with the purpose of our website. I'm sorry to inform you that we will not be taking your idea into consideration during the final phase of our project.

User Story: Graphical Representations of Data

"Hi, I was wondering if y'all could graphically represent the ratings and scores of your different models. For example, The ratings of each amenity could be represented by shading proportions of 5 stars. Also, it would be helpful to use colors to represent the walk score and transit score of a housing instance (>70 could be green, 30-69 could be yellow, and <30 could be red). I think this could really take your website to the next level and make it easier for the me to scan through the website."

Hi @sinemana Thank you for the great ideas! We've come up with something similar but not graphical. We decided to adapt WalkScore's official classification for their scores into 5 categories of plain English terms to help users better understand what the scores mean. Hopefully, that helps you navigate our website.

More information on Animal Friendly Filter

> *"Hello, I am a user looking for an animal-friendly apartment, but my pet is a large dog whose breed is not frequently allowed in apartments. In the Animal Friendly filter, I'd like more information on which animals are allowed at a particular apartment. This will help prevent misunderstandings when looking into animal-friendly apartments, and it will help me narrow my search more effectively."*

We have addressed this request by adding more detailed pet limitation data in our API response body including the number of pets allowed and weight restrictions. However, due to the lack of available APIs, we will not be able to provide breed limitations by building.

Search Filter for Apartments with Accessible Amenities

> *"Hi development team, I am a university student who has some friends with disabilities. I would like a feature that would allow me to filter for apartments that are wheelchair or handicapped accessible. This will be particularly useful for those with limited mobility."*

We have addressed this request by including accessibility features in the building amenity section on our housing pages. We will be utilizing both Apartments.com API and Google Places API to search for accessibility features. The only existing limitation is that Apartments.com contains client property posts that may not include all the accessibility details.

Search feature for Amenities for Apartments

> *"Hi! I'm a user who would like to know about any amenities that an apartment offers. These include free electricity, free wifi, free water, gyms, pool, elevator, roof terrace, etc. While price helps me choose an apartment, allowing me to search for common amenities will make the apartment search a lot easier."*

We have addressed this request by including lists of amenities provided by listed properties in our property pages. Users will be able to search for certain desired amenity options. We have also provided types of utilities that properties will cover.

Safety rating attribute for each student apartment

> *"Hello, I'm a user who heavily prioritizes how safe the area around an apartment is when choosing where to live and to figure this out, I typically have to search a lot on Google for crime statistics and safety ratings of neighborhoods. I was wondering if you could add some more information to the website regarding how safe the area around each student apartment is to ease this process. For example, maybe you could add a general safety rating for each apartment based on this website which has a map showing the safety of specific areas in cities like Austin."*

We have not yet provided support for this request due to the lack of APIs. We plan to address this request by including safety scores on our property pages as demonstrated on our Postman design documentation which we work on for the next phase.

Affiliated Safety Resources for University Students

> *"Hello, I am a user who enjoys going out for entertainment until late in the night. However, I do not have a car, and I am worried about my safety if I have no ride home. Is there a way to check if a certain university has any affiliated safety resources to help me get back home? Some examples of this include UT Austin's Sure Walk system and UT Night Rides."*

We will not be able to address this request because there is no available API for retrieving this information. Finding this information for each school will require scraping over hundreds of official school websites and our team does not have the capacity to write a scraper at the moment.

Campus Gallery

> *"Hi development team, I am a high school graduate looking for Universities. For each of the universities, I'd like to be able to see photos of the campus to grasp the*

situation on hand. It would be great to have a carousel of photos or any other sort of format to display the gallery."

We will have images available for the top 200 schools in the US, available for viewing on the page for each University! We won't be able to display those images on the grid for picking universities during this phase, but perhaps in a future phase we'll be able to add that feature.

University Major Rankings

"I'm a student interested in the rankings of different universities. I know what I want to major in, and I want to know more about how different universities compare for this field. It would be nice if I could see the rankings of universities based on majors."

Major information like ranking is definitely something to consider when applying for a school. However, we will not be able to address your use cases in any upcoming phases. Our team members had to get general ranking info by hand due to the limited availability of appropriate APIs. I hope you can understand our limitations and hope you can keep contributing new ideas that will help build a better platform.

Link to Entertainment instance website

"Hi, I'm a user who is interested in buying tickets or items from different entertainment venues. I was wondering if you could link the venue's website so I could check out their products and prices without having to google them. Thanks!"

We get all of our entertainment information from Yelp API which doesn't include any links to the business itself. We regret to tell you that we might not be able to fulfill your needs during the upcoming phases.

Proximity to current location ranking

"Hi development team, I'm a student looking for something fun to do within the entertainment section of the website, and I want what I'm going to do to be as close to my current location as possible. However, right now the website seems to provide

solely the address of each entertainment, so to find what's closest to me, I must manually check how far each address is on Google maps which is a time-consuming process. Would it be possible to add the ability to sort/rank the entertainments based on how close they are to my current location?"

The idea of getting info based on proximity sure sounds interesting and useful. However, that involves the user granting us permission to get their current location. Furthermore, the purpose of our site is to provide info on entertainment available in locations near the university or housing instead of your current location. I am afraid there is nothing we can do at this point to address your use case.

Contact Information for Apartment

"Hi development team, I am a user who likes some of the apartments that are listed on the apartments page. I would like to contact the apartments that I like, so I can schedule a tour or sign up for a lease. Can you include the phone number or email of the apartment on the apartment's instance page?"

We will try our best to meet your needs during the next phase. We have run out of credits on our API keys to get contact information this month. Hopefully, we will be able to verify if we can make that information available to you.

III. RESTful API

You can find our postman API here:

https://documenter.getpostman.com/view/17627995/UUy3A7Rd

IV. Models

University

- Name (Search)
- Alias (Search)
- City (Search)
- State (Search)
- Zip Code (Search)
- Longitude (Search)
- Latitude (Search)
- Number of Enrolled Students (Sort)
- Ownership (Public/Private) (Filter)
- Acceptance Rate (Filter)
- Graduation Rate (Filter)
- In-state tuition (Sort)
- Out-of-state tuition (Sort)
- Ranking (Sort)
- Average Financial Aid Granted (Search)
- Average SAT scores (Search)
- Carnegie Undergrad (Search)
- School Website URL (Search)

Housing

- Property Name (Search)
- Alias (Search)
- Property Type (e.g. apartment, condo, house) (Filter)
- City (Search)
- State (Search)
- Neighborhood (Search)
- Longitude (Search)
- Latitude (Search)
- Min & Max Rent (Filter)
- Min & Max Bedroom (Filter)
- Min & Max Bathroom (Filter)
- Min & Max Square footage (Filter)
- Pet Friendly (Filter)

- Pet Limitations (Search)
- Schools nearby (Search)
- Building Amenities (Search)
- Walk Score & Transit Score (Search)
- Rating (Sort)
- Accessibility Features (Search)
- Safety Score (Search)
- Included Utilities (Search)

Amenity

- Name (Search)
- Alias (Search)
- Category (Search)
- Rating (Sort)
- Number of Reviews (Sort)
- City (Search)
- State (Search)
- Zip Code (Search)
- Longitude (Search)
- Latitude (Search)
- Age Restriction (Filter)
- Pricing (Search)
- Delivery / Takeout (For restaurants) (Filter)
- Operating Hours (Filter)

Media

University: Pictures, Videos, Map

Housing: Pictures, Texts, Map, Property Description

Amenity: Pictures, Map, Yelp Reviews

Connection

University: University connects to Housing because most universities provide their housing and some people live in housing areas near universities for an easier commute. It also connects to amenities because universities provide students with entertainment,

shops, and restaurant. Otherwise, students will go to these areas near campus because they are more accessible.

Housing: Housing connects to universities because housing is an essential part of universities. You most certainly have to live in the city where the university is located. Housing also connects to Amenities because a lot of places provide their amenities to residents or they have amenities like restaurants and convenience stores nearby.

Amenity: Amenities connects to university because universities provide students with amenities or they have many amenities located nearby. Amenities also connect to housing because many apartments and condos have their amenities. They also tend to have an abundance of amenities nearby.

V. Tools

We used a large variety of tools to make this project possible. Here's a list of tools we used, a link to their original pages, and the purpose we used them for:

- GitLab (https://gitlab.com)
    - GitLab provides us flexible source control, good issue tracking, communication with other teams, and more
- Postman (https://www.postman.com)
    - Postman allows us to manage the API for our website
- React (https://reactjs.org)
    - React provides a more usable framework for front end development
- React Bootstrap (https://react-bootstrap.github.io)
    - React Bootstrap is a CSS framework that incorporates React
- TypeScript (https://www.typescriptlang.org)
    - TypeScript provides a typing system on top of JS, allowing us to be more explicit about how functions work and what to expect from our code
- Namecheap (https://www.namecheap.com)
    - Namecheap allowed us to procure a domain for our website
- Discord (https://discord.com)

    - Discord allows us to have very active discussions and voice chats to keep all the members of our team as informed as possible.
- Amazon Web Service (https://aws.amazon.com)
    - Amazon Web Service helped us host our GitLab repository online and even incorporates live redistributing new code
- NodeJS (https://nodejs.org/en)
    - NodeJS provides a more useful JS runtime for us to experiment with our changes on
- Node Package Manager (https://www.npmjs.com)
    - NPM allows us to easily manage dependencies and packages for our code
- Jest (https://jestjs.io)
    - Jest allows us to write effective unit tests for front-end components
- Axios (https://axios-http.com)
    - Axios provides an API to streamline and simplify our GET requests
- Selenium (https://www.selenium.dev)
    - Selenium allows us to run effective end-t--end and acceptance tests
- Python (https://python.org)
    - Python provides a deep and useful language to manage our back end functions
- NGINX (https://www.nginx.com)
    - NGINX allows us to host our back-end servers
- AWS Elastic Beanstalk (https://aws.amazon.com/elasticbeanstalk)
    - Elastic Beanstalk helps us manage back-end server deployment
- Docker (https://www.docker.com)
    - Docker allows us to create portable containers for our front-end and back-end
- Amazon RDS (https://aws.amazon.com/rds)
    - Allows us to create a manageable relational database
- Marshmallow (https://marshmallow.readthedocs.io/en/stable)
    - Allows us to serialize python objects and convert them to JSON
- Flask (https://flask.palletsprojects.com/en/1.1.x)

    - Provides a framework for our Python web development
- SQLAlchemy (https://www.sqlalchemy.org)
    - Provides a toolkit for running SQL queries in Python
- Different API and Data sources:
    - https://www.yelp.com/developers/documentation/v3
        - Amenity reviews
    - https://collegescorecard.ed.gov/data/documentation/
        - All data for colleges apart from ranks and images
        - The College ScoreCard API provided a huge list of values so we had to decide which fields are relevant. This requires some digging through their data dictionary.
    - https://apify.com/tugkan/apartments-scraper#apartments-scraper
        - Data for housing
        - There was no publicly available API for scraping housing information as thorough as the scraper found on APIFY. APIFY will execute the scraper and store the results into their data storage. Then we can use their RESTful API to get the data stored in APIFY storage.
    - https://developers.google.com/maps/documentation/places/web-service/overview
        - Images for the models
        - Videos for universities
    - https://docs.gitlab.com/ee/api/api_resources.html
        - GitLab commit and issue information

VI. Hosting

FRONTEND

We used NameCheap to obtain our domain name campuscatalog.me. We hosted our frontend using AWS Amplify, which syncs to our GitLab repository and redeploys when any changes are made to the production branch.

BACKEND

We then deployed the Flask app to AWS. We started by setting up development and production Docker images. Next, we deployed to elastic beanstalk by creating an environment and application. Finally, we got an AWS certificate and connected it with the DNS.

VII. Phase II Features

The primary features of phase 2 were the database population, the API, and the pagination.

- **Pagination** was implemented on the front-end. Whenever a user accesses a model page, the page requests every instance of that model. However, the table/grid built from that information only shows a small slice of the entire response. There are buttons which, when clicked, change the portion of the response that is sliced. This means that the model pages may take some time to load initially, but will not run a new query when the page is changed.
- For our database, we decided to use **PostgreSQL** provided by AWS because that's the one our phase leader is most comfortable with. The details of implementations are visible in the UML diagram we posted. All three models are linked to each other by their locations. Each model also has their own children tables with more images.
- We implemented our **RESTful API** using Flask and Marshmallow with the support of SQLAlchemy. SQLAlchemy allows us to establish connections to our Database and execute queries using python. Marshmallow enabled us to generate prettier JSON results with ease. Flask is used for routing to different endpoints.

VIII. PHASE III FEATURES

The primary features of phase 3 were implementing searching, sorting, and filtering, and pagination on the backend and then creating frontend features to go along with them.

- During this phase, there were a couple of features we needed to implement on the back-end.
    - Pagination: In the previous phase, we enabled pagination on the front-end. This phase, we used the Pagination object provided by SQLAlchemy for easier pagination. We added a header in our JSON responses for the front-end. The header includes the current page, number of items per page, max number of items, and max number of pages. This allows the frontend to manipulate the results and display them accordingly.
    - Sorting and Filtering: We added new parameters to our endpoints for searching and filtering. Sorting accepts the column name and ordering. The filters are enabled using the filter function provided by SQLAlchemy.
    - Searching: We implemented naive searching which tries to return results that match every term in the search query.
- For the frontend search, we added the Search Bar to the Navbar. The Navbar is displayed on all of the pages of the website. Once you search in the search bar, it takes you to a separate search page. On the search results page, there are three clickable boxes which represent each model. By clicking on the models, we can filter the search results. The results are displayed in cards which display different attributes of the instance. After clicking on the instance, the user is taken to the instance page. The part which is searched is also highlighted (ctrl-f style). To implement the search frontend, we continued using **React** and different **Bootstrap** elements.
- For frontend sort, we added an onClick functionality to the column headers of the tables and a list of radio buttons to sort the grid. Clicking one of these column headers on a table will first sort by that column ascending, then sort by it descending, then finally remove all sorts. This is managed by updating the query string and reloading our results. The radio buttons on the grid work similarly in

the sense that clicking any of them will update the query string and therefore reload the page.

- For the frontend filter, we added a popover which appears on clicking a button. This popover will display a list of options for users to specify what types of outputs they wish to see. Whenever the user clicks “Apply”, those user-specified params are built into a filter string which is then used to change the query. The new query is run and then the results change.

IX. Phase IV Features

Choropleth

- This graph is used to show the total number of universities, housing options, and amenities in each state (according to our database). It uses some publicly available libraries and files to draw a map of the United States, then fills in each state with a color. This color is calculated on a linear scale depending on how many of each model that state has (so a state with more universities will become a darker color). There’s also a tooltip to see the state’s exact number. We added a drop down menu to freely switch between the models, showing either universities (in green), housing options (in red), or amenities (in blue).

  This component uses d3 directly (appending to an SVG). It runs two effects: one to query our database and parse the data (which runs exactly once), and one to draw the chart (which re-runs every single time you switch models).

Bubble Chart

- This graph is a simple bubble chart that shows the price of each university. The size of the bubble is proportional to the cost of tuition. It works by running a single query to our back-end, parsing and formatting the data, and then using a library to create the chart itself.

Scatterplot

- This graph is a recharts scatterplot. It plots housing options, showing their walk score vs their average rent and their transit score vs their average rent. You can switch between these two options with a drop down menu. Much like the

choropleth graph, it uses two effects: one to get data (once), and one to redraw (every time you change the score you're viewing).

Sunburst

- The first of our provider visualizations. This queries every single course, department, and professor from their database and uses it to build a zoomable sunburst graph using a library on top of d3. The sunburst graph is a hierarchy of departments which contain professors which contain classes which have a number of sections. The actual number of sections is so large that it basically crashes any browser, so we filtered out to only the departments with over 100 sections to keep the graph lean. You can click on any department, professor, our course to zoom in and get a better look at that sub-section. Clicking on the middle will zoom back out.

Sankey Graph

- We queried all the courses at UT and built a Sankey chart showing the scheduling of these classes. Every course has a schedule and each schedule is composed of some days (ex: MWF is a schedule, and [Monday, Wednesday, Friday] is the days). We show how many courses have each schedule, and then how many courses are on each day in a simple flow. There are also labels and tooltips to simplify and explain everything. This component is also built using recharts, and runs exactly one query to the provider database.

Provider Scatterplot

- A very simple recharts scatterplot that just shows the number of courses vs the number of professors in each department. We query their departments and then format the data and pass it along to the component.

Refactors

- FRONT-END
    - Consistency: Standardized strings and colors across the site
    - Styling: Add CSS to help everything look better and be more readable
    - QoL: Added features to make navigation and pagination more intuitive

    - Reusability: Broke large components apart, and re-wrote very similar components into more complex components that are reused
    - Documentation: Added comments to frequently explored areas and critical portions of the code to help others who need to write similar things.
- BACK-END
    - Modularization: Following best practices described on Flask documentations, we made our main application into a module and separated unit testing into a different module.
    - Exception Handling: There were some exceptions in SQLAlchemy that weren't handled properly. We also added custom exceptions to make our exception messages easier to understand.
    - Reusability: There are a bunch of helper methods that serve the same functionality for different models endpoints. Merging them together makes the code more readable and makes the functionality more scalable if we were to add more models.
    - Readability: Our endpoint file contained a bunch of helper and handler methods. To improve readability, we moved it into a separate file.

X. GITLAB

You can find our repository here: https://gitlab.com/RG8452/campus-catalog