

CSE 457A Project Proposal

Basic Info. *The project title, your names, e-mail addresses, IDs, a link to the project repository.*

Title: Visualizing UCLA graduate admission.

Names: Jordan Chisam, Steven Harris, Zhengliang Liu

Emails: jachisam@wustl.edu, sharris22@wustl.edu, zhengliang@wustl.edu

IDs: 441729 (Jordan), 393270 (Steven), 465270 (Zhengliang)

Repository: <https://github.com/jachisam/cse457a-finalproject>

Background and Motivation. *Discuss your motivations and reasons for choosing this project, especially any background or research interests that may have influenced your decision.*

College admission statistics, including those for graduate schools are always of great interest and importance for young people who are aspiring for getting better education. We have all gone through numerous applications. Our portfolios have all gone through numerous admission committees. Even though an applicant could evaluate her or his potentials for admissions, for the most part the admission process is a black box that provides little feedback. Finding out what factors lead to admission is certainly an important research topic with practical impacts. It would be very helpful if one could look at past admission data and make an assessment before investing the energy, time and money on an application for a particular school. Sadly, most schools never disclose such data. For those that do, the data might not be intuitive to be understood, More importantly, it is difficult to understand how exactly the measurements for each of the metrics factor into the final admission decision. Thus, we are very excited to start this project to visualize admission data with the aim of building an intuitive representation of the data. We found a dataset from UCLA on [Kaggle](#) and we think it is a good start, especially given that there is such a scarcity of admission data and even less visualization with such data.

Project Objectives. *Provide the primary questions you are trying to answer with your visualization. What would you like to learn and accomplish? List the benefits.*

How to effectively visualize each factor's contribution to the application process? With so many factors of consideration in the application process, it would be nice to be able to visualize each one of them clearly and effectively.

Benefits:

- Provide mechanism to compare and contrast one's performance as compared to their peers on a factor-by-factor basis.
- Provide feedback for self assessment and allows students to understand where they fit in the academic spectrum.
- While college selection isn't so cut and dry, it can be a good indicator of realistic options for future schools as well give some indicators on future academic achievement. Essentially, a good visualization can help students break down the obscure admissions process into concrete information of students' performances in relevant metrics.
- Helps potential master students measure their chances of getting into a particular university. We, the developers could also learn something about a good school (e.g., UCLA)'s admission process.

How to intuitively convey the potential admission result? Instead of handing out raw data in row-column tables, will it be more intuitive and understandable to visualize the final admission result? In our dataset, the original authors provide their machine learning analysis results to indicate the probability of admission for a particular student. We plan to label the ones with >80% probability as "admitted" and the rest as "not admitted". We want to use an intuitive visualization to convey this result to users.

Benefits:

- Intuitively convey the final result so users don't need to dig into the rows and columns unless they really want to.
- Make the data interpretation process more fun and exciting.
- For us, we can sharpen our skills of using D3 to build a visualization that is intuitive, informative and useful.

While we may not be able to fully formulate those questions until we have a solid data presentation, it will be interesting to see what relationships emerge once we have a proper data presentation.

Data. *From where and how are you collecting your data? If appropriate, provide a link to your data sources.*

We find our dataset on Kaggle. It is collected by a person named Mohan S Acharya, part of his paper on a relevant topic [1]. Link:
<https://www.kaggle.com/mohansacharya/graduate-admissions>.

[1] Mohan S Acharya, Asfia Armaan, Aneeta S Antony : A Comparison of Regression Models for Prediction of Graduate Admissions, IEEE International Conference on Computational Intelligence in Data Science 2019.

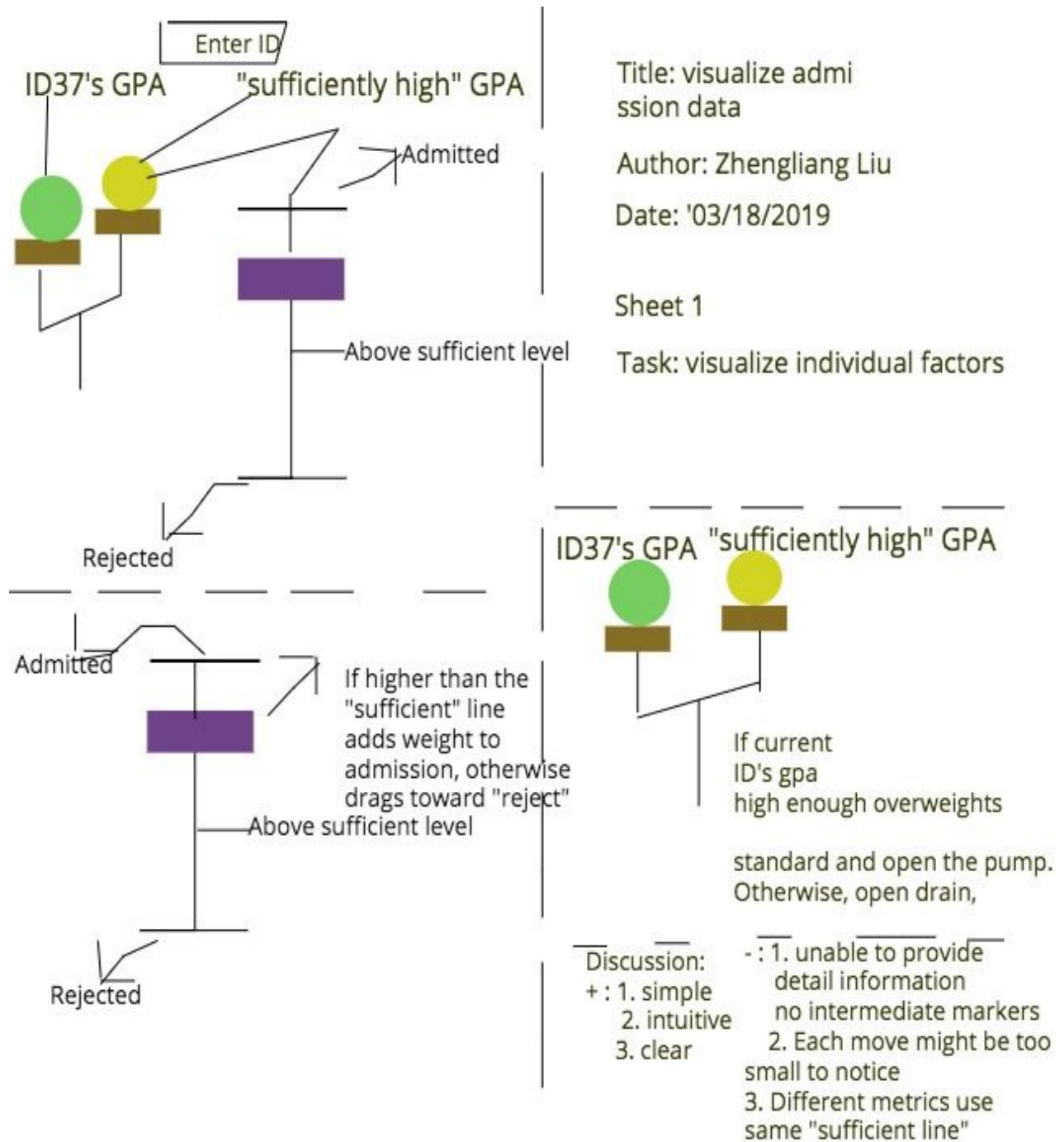
Data Processing. *Do you expect to do substantial data cleanup? What quantities do you plan to derive from your data? How will data processing be implemented?*

No, we do not expect to do substantial data cleanup. The dataset is very clean in the first place, nor is it particularly large. Data processing will likely be implemented with bash scripts or Python, or both.

Visualization Design. *How will you display your data? Provide some general ideas that you have for the visualization design. Create **three alternative designs for your visualization**. Create **one final design that incorporates the best of your three designs**. Describe your designs and justify your choices of visual encodings. You use the Five Design Sheet Methodology.*

Alternative Designs:

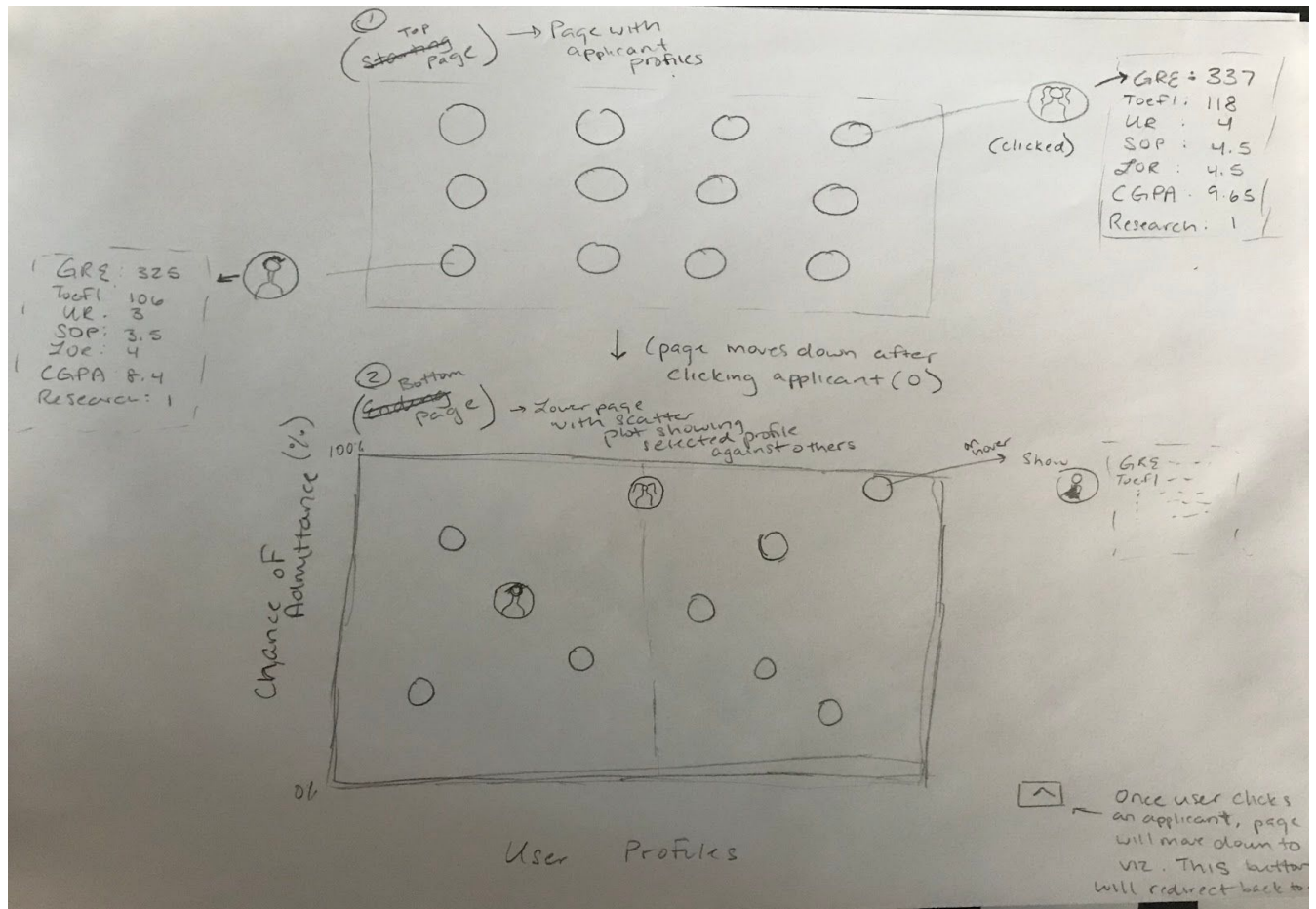
(1)



I use a "balance" to weigh a factor (e.g., gpa = 3.74) against a pre-calculated "sufficient level" (e.g., "3.8"). The size of the ball represents the value of the factor (such as "3.74"). If the balance favors that ID (in this case 37) 's GPA, it drags the bar upwards

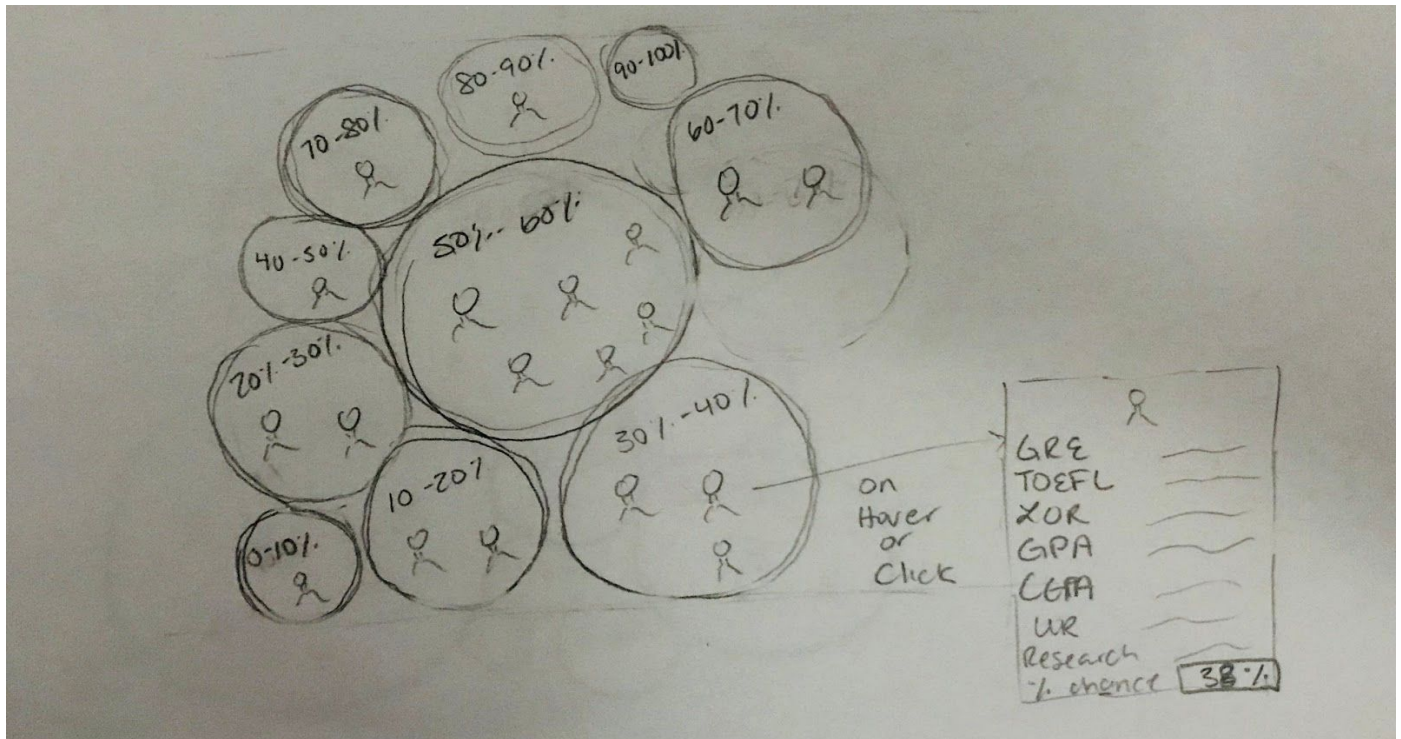
toward the line marked as "admitted", otherwise it loosens it down to get closer to the line marked as "reject". Gradually, as we go through every factor (GPA-->TOEFL-->GRE-->recommendation letter etc) it moves toward either rejection or admission. The strings between the balance and the purple bar acts as strings of control. The color of the objects are for pop-out effect only with no special meanings, for now. It is simple and intuitive but the largest problem is that. The key is, with this design it is harder to differentiate the different importance of two different factors (e.g., GPA v.s. GRE score) because every factor shares that "sufficient" marker and it is hard to differentiate the effects visually.

(2) This is a very rough sketch of a *basic* single page visualization that allows users to look at various graduate school applicant profiles. Each profile will show the variables that are taken into account during the graduate school admissions process. As the user clicks on a circle (representing an applicant profile), the page will move down and display a scatter plot with the selected user profile (highlighted and larger) compared with other user profiles (non-highlighted and smaller--to avoid clutter). On hover, the circles (profiles) will enlarge and show their respective variables.



The downside of this visualization is that it is not clear how certain variables affect chance of admission. Additionally, it may be a bit monotonous for the user to go back and forth from the top of the page to the bottom. Our final design should take these faults into account and make it clear which variables have stronger influence, in addition to making user interaction easy rather than tedious.

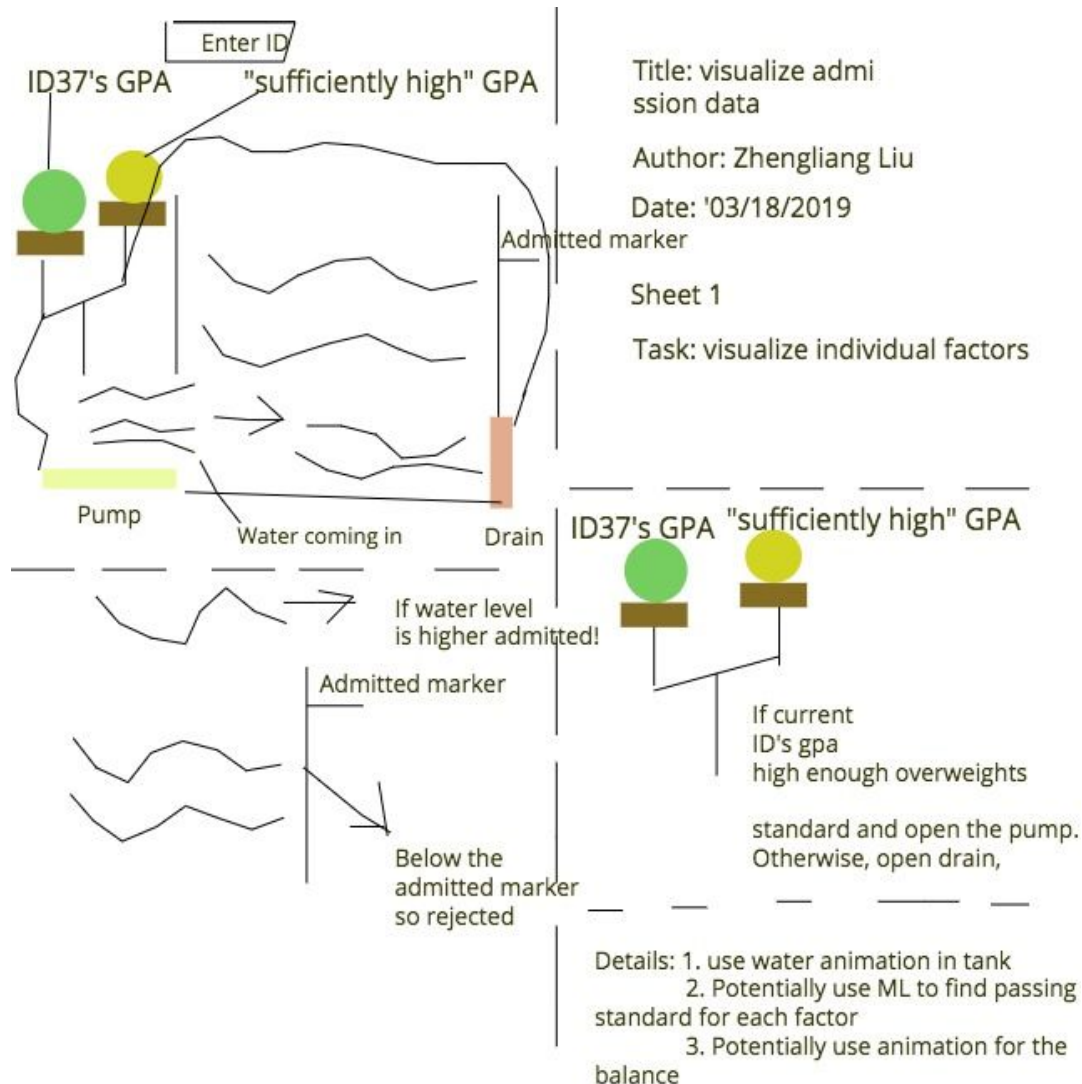
(3) Another alternative design was to use a bubble graph visualization. This would depicted all of the probabilities of admission in 10% ranges. The lower part of the visualization would be used for clusters of low chances of admission, while the higher part of the visualization would be used for higher chances of admission. The size of the bubbles would demonstrate the portion of applicants that fall into that range. Each bubble would contain various applicants that fell into the chance of acceptance rate represented. On hover (or click), the variables of



each application would be displayed in a tooltip.

Ideally, this visualization would use color to separate the circles. Similar to (2), this visualization lacks the ability to properly demonstrate how each variable affects chance of admission.

Final Design:



The final design is an improvement of the first design with the “balance” and the “hanging bar”. It uses a water tank instead. In the above example with GPA, if the balance favors ID 37’s GPA against a pre-calculated standard/”sufficient” level, it opens the gate of the water pump and let water in. Otherwise, it opens the gate of the water drain and drains some water out. Gradually, as we go through every factor (GPA-->TOEFL-->GRE-->recommendation letter etc) the water level goes up and down and eventually if it goes beyond the “admitted marker” this ID is admitted, otherwise rejected. The main improvement from the “hanging bar” design is that with water tank you can fill different amount of water to compare the importance of different factors (GPA vs GRE, the more important the more water coming in) but with the “hanging bar” it is harder to do that. Also, potentially, as the water tank design is inherently slightly

more complex it can later convey more information from the each run's water feature (color, "wave") for more information. It is more expandable and flexible.

Must-Have Features. *List the features without which you would consider your project to be a failure.*

- Must visualize the applicant's chance of getting into the university
- Users should be able to select potential applicant profiles to test with the visualization
- Must be able to visualize the weight and impact of each factor individually
- Must be interactive

Optional Features. *List the features which you consider to be nice to have, but not critical.*

- Users can input their own variables(scores, rating, GPA, etc.) to determine their chances of getting into the university
- Potentially, if we could find more data or data organized by years, we could add visualization for the trends.

Project Schedule. *Make sure that you plan your work so that you can avoid a big rush right before the final project deadline, and delegate different modules and responsibilities among your team members. Write this in terms of weekly deadlines.*

Weekly Deadlines (mondays)

March 18 - Submit Project Proposal

March 25 - Together work towards completing 1-2 feature that are necessary to demonstrate project functionality for milestone 1 (ex: visualize chance of acceptance at a basic level--not all transitions/design need to be implemented at this point)

April 1 - Submit Milestone I with functional prototype (must have at least one fully functional feature that demonstrates visualization goals)

- After this date, *evenly* split the "must have features" amongst the three group members and begin to work

April 8 - Touch base with other group member and see what progress everyone has made on their assigned feature(s)

- Assist teammates who need help completing their features or provide advice for polishing other features
- Alternatively, begin working towards optional features

April 15 - Submit Milestone II

- Complete all of the listed “must-have features” by this date.

April 18 - Complete User Studies in class (Thursday)

- Follow up: Revise from user feedback and debug the prototype

April 21 - All features/designs must be implemented and reviewed before today (Wednesday)

- Rehearse demo of the visualization today and pre-answer* questions listed for the project presentation:
 - What do you feel is the best part of your project?
 - What insights did you gain?
 - What is the single most important thing you would like your audience to take away?

* Each team member will take one question--or two team members will talk while one person demos the actual functionality

April 22/25 - Final Project Presentation (Tuesday/Thursday)

April 29 - Submit Final Project and Peer Evaluations