

Team Number: 23

Team Name: LeGroup

Github Repo Link: <https://github.com/cse110-sp24-group23/warmup-exercise-team-1>

Youtube link: <https://youtu.be/zcSOFv9NPA8>

## SWOT ANALYSIS:

Since we have 11 people, we decided to split into two groups so that all of us have a chance to create meaningful contributions. While both teams chose to implement the sentiment tracker, we chose different methods of workflow to see which would work better.

To start out, Group 1 met up to talk about wants/needs for the app, features that could fulfill those, and researched what other people have done when it comes to rating widgets. We created a Miro board to keep track of user stories, research, ideas we liked, how to improve on them, and ultimately the steps it would take. Once we had our users wants/needs, we kept track of the common themes and tried to fulfill them without overcomplicating the design. From there, we broke the project down by features/problems, tests, docs, and more. We listed them out and created github issues with labels based on difficulty and issue progress. Each person chose one hard/medium difficulty issue and one easy issue that they felt comfortable with. We set soft and hard deadlines to keep us on track and specified which issues were urgent due to them being prerequisites for other issues. After everything was done, we had a well functioning, accessible, and clean app.

We were able to quickly collect research and decide on a design, altering ideas to fit our needs and avoid user confusion. Since people were able to pick what GitHub issue they wanted to tackle, everyone was able to step into roles they were comfortable with. Also, the team was able to work out smaller interpersonal issues in a productive way that ensured psychological safety.

However, this method made it hard to keep track of progress. While some tasks were simple and could be finished quickly, they often relied on other tasks to be completed first, resulting in dependencies that led to a backlog of several issues. Also, because we only had a week to work, and the widget wasn't very complex, it was difficult to equally distribute the workload. People were unable to take on roles that involved learning new material.

Breaking down larger problems into smaller tasks provided an opportunity for people to ask teammates for guidance. We also have the opportunity to try and implement more regimented workflows, especially as we continue to learn how to use more tools.

At the beginning of the week, Group 2 met up to brainstorm ideas for the project. We started by identifying what type of sentiment tracker we wanted to make. We decided on making a mood tracker for mental health purposes. After deciding our theme, we thought of different user stories and listed out features that would be beneficial to various types of users. For example, we deliberately chose to make selectable buttons for each emotion, and then have a

separate submit button so that users could reflect on their emotion and then lock in their response when they were satisfied with their selection. This way, users who may be more indecisive could have a chance to gather their thoughts. Then, we decided on a general layout. We grouped our tasks into labels of HTML, CSS, and Javascript, and created issues within those labels. Each person chose what language they would feel comfortable working with. We set up issues in our repository that reflected the tasks that needed to be completed, and set up deadlines so that we could stay on track. By the end of the week, we had a solid project that was aesthetically pleasing, had accessibility features, and was well tested.

Because of this workflow, everyone knew what they needed to do. Our branches and issues were set up well and it made it easy to resolve issues by pull requests. The work felt evenly divided, and as new tasks popped up, we were able to share responsibilities by communicating as soon as possible. The deadlines we set for ourselves created streamlined expectations that people could accomplish.

However, time was a big concern for us. Many of us were not very well-versed with development with GitHub, and we spent a lot of time learning what features to use and how to use them. Furthermore, we didn't include much Javascript since we were all learning how to use it for the first time. There were other small weaknesses, like forgetting to add comments or close issues, but these will most likely get worked out as we get more practice.

This experience gave us the opportunity to learn how we work together as a team, and what expectations we have from each other. Moving forward, we have a more clear idea of what needs to be improved and where we need to improve our individual skills. We realized that there are many tools we should learn how to use, like github actions and pipelines, in order to help us streamline our process.

While we didn't run into any major issues, our workflow was very heavy on dependencies. Without the HTML finished, the CSS and JavaScript teams couldn't do much. While this worked out for us because we planned our time effectively, this may pose a threat to a larger, more complicated project. To continue working in this format, we would need to have a very structured approach with deadlines to ensure everyone is on track. Furthermore, since we didn't have enough experience building software, we were very worried that we might have overlooked some critical concepts.

In the future, we will meet as a team to discuss the benefits and deficits of each method in order to decide how we'll tackle the development process of our next project.