Team Members: Leo Li & Tracy Yang
Project Name: Cry Together— Crying Room
Project Goal:

We will develop a website where users can join when they are feeling blue and need a room to cry in. It is a cross-cultural norm for humans to disguise vulnerabilities in public places and in certain social occasions. This leads to quiet sufferers of sadness, loneliness, and depression. We will mitigate this epidemic by providing an online platform that acts as a safe space for people to unite under the universal experience of negative emotions. This website allows users to log on and gain instant mood relief whether they are having a severe mental breakdown or subtle melancholia.

When users enter the website, they will be directed to a crying room specific to their continent. This will require the integration of geolocation tracking API, privacy settings, and possibly storing user information in a database(?).

Subsequently, this will enable a load balancer to distribute users to different servers based on information on their geolocation. We will need to implement the load balancer to handle incoming requests and distribute them to several simultaneously running servers.

When users enter their room, the center of the screen will show a large rotating ball that displays the number of current cryers in the room. The number is real-time updated: it changes as users join or leave without the client refreshing. This requires the server to handle multiple concurrent connections and push updates to clients whenever there is a change in online user count: using the mechanism of WebSockets vs. server-sent events.

Finally, we will test our application by simulating 1,000- 10,000 users coming into the crying room.

Plan:

Now until 4/11:

- build a basic interface and a simple client-server connection. Refer to the in-class work we did in module 1.
- build a pro-type load balancer that randomly distributes incoming requests, with manually defined maximum capacity.
- Think of potential problems we might encounter for the 4/11 meeting.

4/11 - 4/18:

- Build multiple servers, and connect them to the load balancer.
- Test and test and test.
- If we have time, start looking into geolocation tracking.

4/18 - 4/22:

- If time allows, implement geolocation to replace random assignment in the rooms.
- Finalize slides for presentation.