

Proposal!

I would like to create a social media application called *CallingOut*, where people can express their frustrations and improve relationships by openly addressing issues before they escalate. This app will help users "call out" friends, family, partners, and so forth, aiming to reduce resentment, help to get closure, and keep relationships smooth and healthy.

So the idea of *CallingOut* is to allow people to communicate their issues directly, honestly, and at the right time. Instead of letting anger or frustration build up, users can send a message to someone they're upset with, letting them know what's bothering them. This helps users let go of grudges and address issues head-on, promoting honesty and clarity. The goal is to help users gain closure, avoid resentment, and save time that might otherwise be spent dwelling on unresolved feelings.

Technology Stack:

- **Front End** → HTML, CSS, and JavaScript using AJAX to communicate with the backend—I am familiar with HTML, CSS, JavaScript, and AJAX.
- **Backend** → Java with the framework of Spring Boot for handling requests
- **Database** → MongoDB for storing user data and messages user details, callouts, replies, and reactions

In the *CallingOut* application, I might have the following interfaces, parent classes, and their children, and I would be able to use the filter, fold, and lambda expression, as well as recursion.

Classes are following : User Class, UserAction Interface, Callout Class, Reply Class, Reaction Class, Notification Class so forth.