Front-End task: GeoChat

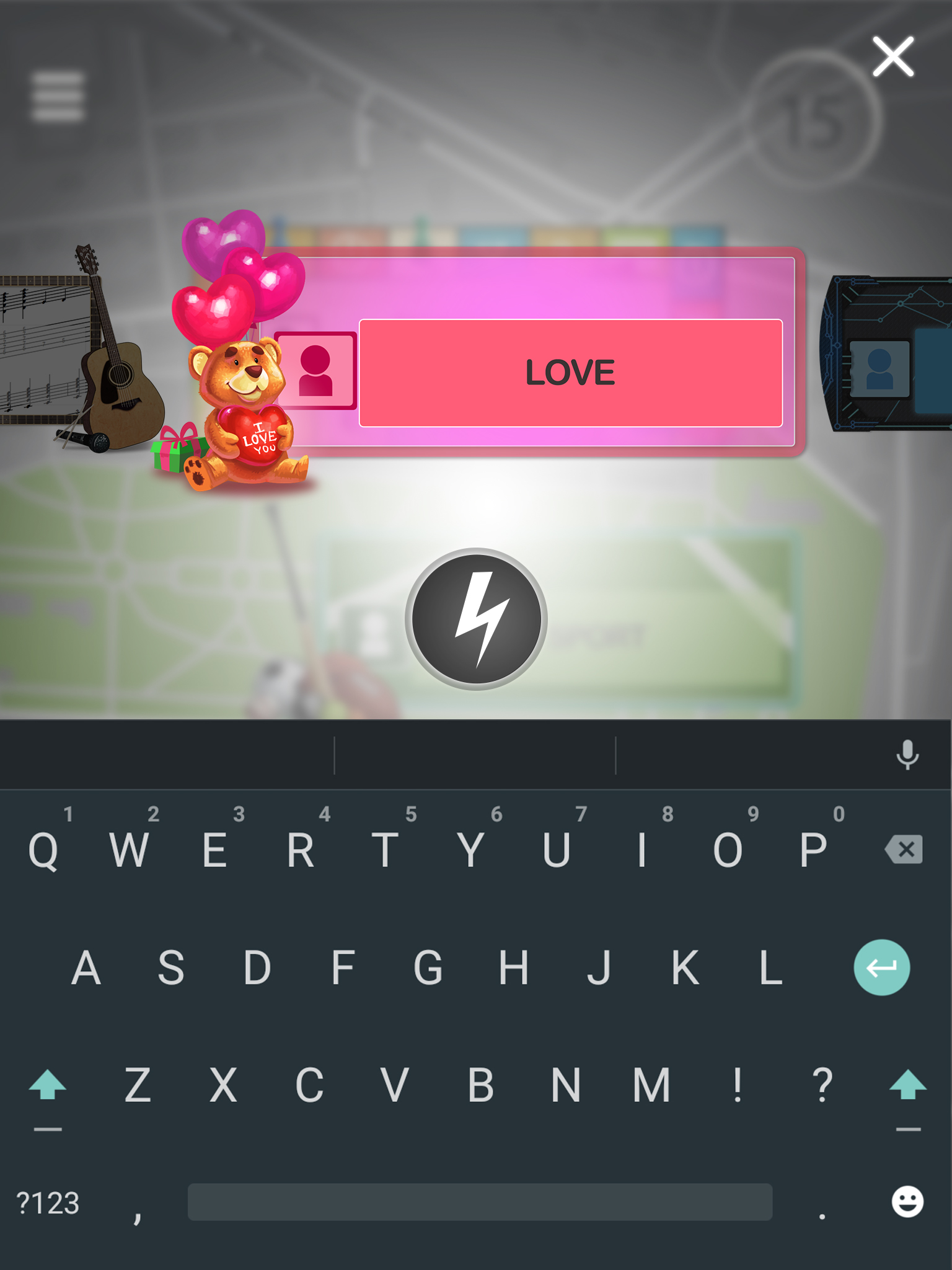
Create an application that looks as closely to the designs as possible and functions as described below, don’t develop a supporting backend, only use external SaaS services like <https://www.mapbox.com/>

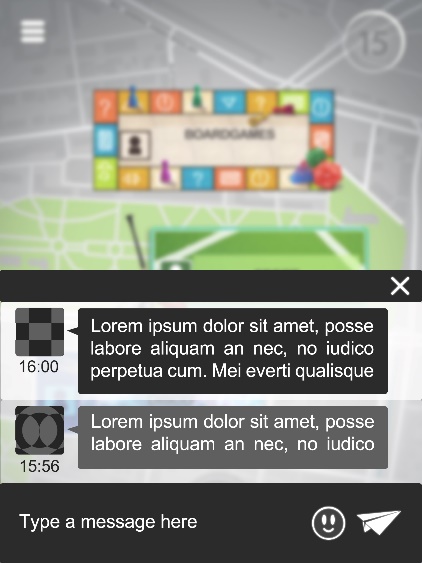
Note: use the resources from the Screens and ShoutDesigns folders for the visuals.

Client Story: The client wants us to create a geo chat application, in which the users are going to be able to create GPS positioned message boards called Shouts. The client wants to limit the amount of Shouts the users are going to be able to create, to limit spam on the platform, each user is going to start out with 15 points to spent and each created Shout is going to cost 3 points. One of the USP’s of the app is the ability to choose a thematic background for the Shout, the users must be able to select from a list of unlocked backgrounds. For the MVP the client does not want to have an actual backend so all state must be only stored client side and data to be mocked.



1. Main View
   1. When the user opens the app they need to see a map centered on their location and showing all preexisting Shouts
   2. The UI must have a way of creating a new Shout using the button on the bottom, each shout costs 3 points, if the user does not have enough points to create a Shout hey need to be informed.
   3. Each time the user creates a Shout the counter on top must decrement by 3
   4. Each Shout has a place for the Title and the user icon of the user that created it



1. Shout Create View
   1. The user must be able to choose the background for the new Shout by sliding left or right
   2. The Title of the Shout must be displayed directly on the Shout
   3. The user must be able to confirm or cancel the creation by clicking on the “Thunder” button or “X” button respectively
   4. Creation of the Shout happens on the current user GPS location
2. Comment View
   1. The user must be able to click on an existing Shout and see the comments on it
   2. Each comment contains [BODY], [AUTHOR ICON], [TIMESTAMP]
   3. The user must be able to comment by writing in a text field on the bottom, when the comment is sent it must show in the comments list on top with the appropriate user icon and timestamp