---

layout: post

title: BlocChat

thumbnail-path: "img/blocchat.jpg"

short-description: Build a chat room using Bootstrap and Firebase for users to be able to create new chatrooms and send messages.

---

{:.center}

![]({{ site.baseurl }}/img/blocchat.jpg)

## Explanation

BlocChat is a module formed by Bloc, which provides the developer in training a chance to build an app with little to no supervision by a mentor. The functionality of BlocChat is to provide a user with an application that allows them to message another user, create a username and new chat room.

## Problem

I was given this project to expand my experience using AngularJS, UI Bootstrap, and Firebase. I ran into problems initially because I did not readily see the association between the Model-View-Controller. Though textually I understood the relationship between the three, I did not have a solid understanding of "this" and "$scope" in AngularJS controllers. Initially I was thinking $scope and “this” referred to the same instance, but when trying to relay data from the view to the controller, I was running into errors.

## Solution

For this problem, my solution was to conduct more research to better understand AngularJS’s difference between the two. “This” and $scope in AngularJS can refer to the same instance/controller/constructor, but more often than not $scope refers to the isolatescope created when the controller function is run. In AngularJS, the $scope variable is the only reference to the controller that communicates with the view. So, if I wanted to pass data, say from an ng-click event, I needed to establish the $scope reference in the controller function.

-References:

StackExchange: <http://stackoverflow.com/questions/11605917/this-vs-scope-in-angularjs-controllers>

AngularJS Documentation: <https://docs.angularjs.org/guide/scope>

## Results

While writing the code, I had used “controller as” and var x = this, to help mitigate the errors in functionality. As the app progressed and came closer to completion (stylistically it is still a work in progress) I gained a better understanding of the use for “this” and “$scope” in regards to AngularJS, and ended up removing the excess assignments to this and utilized $scope more efficiently. In my homt.html, I did use ng-controller=”xCtrl as $y” but that was because the page is controlled by a different controller, I just needed this one portion of the view to be controlled by separate controller.

> “The functionality is good, but the presentation of the app could be more visually appealing! Overall the chat app works as it was designed to!” - Jennifer Washburn

## Conclusion

Using $scope in the controller and associated functions in the view allowed me to pass the user input data into the model’s controller. Associating “this” to a function that required communication from the view, did not work, and thus user input was not passed to the controller and the subsequently the view could not be updated appropriately.

Going into the project I was very confident that I had absorbed my prior training well, but upon working through each checkpoint, my lack of knowledge was highlighted! This project has been a great asset in my training, because it required me to gain the understandings that I was lacking without being able to rely on someone else to give me direction (lazy thinkers easy way out ☺). The research and time put into this project has been an absolute joy for me, though at times I felt like I was beating my head against a wall, I loved solving problems, adding to my base knowledge, and researching for the answers.