Real time chating app using ionic

1...import firedatabase first in app.module.ts

import { AngularFireModule } from 'angularfire2';

import { AngularFireAuthModule } from 'angularfire2/auth';

import { AngularFirestoreModule } from 'angularfire2/firestore';

import { AngularFireDatabaseModule, AngularFireDatabase } from 'angularfire2/database';

const firebaseAuth = {

apiKey: "AIzaSyAqpzoyp7Djc2SbtjYR7alOfFcHshWo\_tc",

authDomain: "[testproject-7e0dd.firebaseapp.com](http://testproject-7e0dd.firebaseapp.com/)",

databaseURL: "[https://testproject-7e0dd.firebaseio.com](https://testproject-7e0dd.firebaseio.com/)",

projectId: "testproject-7e0dd",

storageBucket: "[testproject-7e0dd.appspot.com](http://testproject-7e0dd.appspot.com/)",

messagingSenderId: "807736096233"

};

@NgModule({

declarations: [

MyApp,

AboutPage,

ContactPage,

HomePage,

TabsPage,

ChatPage

],

imports: [

BrowserModule,

IonicModule.forRoot(MyApp),

AngularFireModule.initializeApp(firebaseAuth),

AngularFireAuthModule,

AngularFirestoreModule ,// imports firebase/firestore, only needed for database features

AngularFireDatabaseModule

],

2.transfer username one page to another y params

username:string='';

....

if(/^[a-zA-Z0-9]+$/.test(this.username)){

this.navCtrl.push(ChatPage,{

username:this.username

});

}else{

this.showAlert("Error","not Formated string");}

}

.........retrive data.................

username:string='';

username:string='';

....

username:this.navParams.get('username')

3.send data to fire base

clickedsend(){

this.firedb.list('/chat').push({

username:this.navParams.get('username'),

msgChat:this.msgChat

})

}

4.retrive data from firebase

username:string='';

msgChat:string='';

message:object[]=[];

..........................

this.firedb.list('/chat').valueChanges().subscribe(data =>{

console.log(data);

this.message=data;

});

in html

<div \*ngFor="let message of message" id="msgchatbody">

<div class="username" >{{message.username}}</div>

<div class="message" >{{message.msgChat}}</div>

</div>



Check for attachment