



socket.io express - socket io, node js, Simple example to send image/files from server to client



Voted #1 Helpdesk Software
Freshdesk

Go where your customers are. Provide exceptional support through social media.



 1 Answers

Send image using socket io in ios

for example, you can send a image like this

```
//emit filename and size to server
```

```
- (void)testButtonClicked {

    _fileName = @"001@3x.png";
    NSString *path = [[NSBundle mainBundle] pathForResource:@"001@3x" ofType:@"png"];
    _imgData = [NSData dataWithContentsOfFile:path];

    float length = [_imgData length];
    NSString *fileSizeString = [NSString stringWithFormat:@"%f", length];

    NSArray *array = [NSArray arrayWithObject:@{@"Name": _fileName, @"Size":fileSizeString}];
    [socket emit:@"start" withItems:array];
}
```

```
//server callback
```

```
[socket on:@"moreData" callback:^(NSArray *array, SocketAckEmitter *emitter) {

    NSString *dataPosition = [array[0]objectForKey:@"position"];

    [self uploadAvatarImage:dataPosition];
}];
```

```
// uploadImage Method
```

```
- (void)uploadAvatarImage:(NSString *)dataPosition {

    NSString *path = [[NSBundle mainBundle] pathForResource:@"001@3x" ofType:@"png"];
    _imgData = [NSData dataWithContentsOfFile:path];

    NSString *imageString = [[NSString alloc] initWithData:_imgData
encoding:NSUTF8StringEncodingAllowLossy];

    NSRange range;
    if ([dataPosition intValue]<[_imgData length]) {

        if ([_imgData length]-[dataPosition intValue]>5120) {
            range.length = 5120;
        }else {
            range.length = [_imgData length]-[dataPosition intValue];
        }

        range.location =[dataPosition intValue];
        NSString *subString = [imageString substringWithRange:range];
        NSArray *array = [NSArray arrayWithObject:@{@"Name": _fileName, @"Segment":subString}];

        [socket emit:@"upload" withItems:array];
    }
}
```

? upload chat room

Is there any plain and straight forward examples on how to serve an image? from server to client? through buffering or just a direct call to download? (the goal is to get image files in near real time efficiently to sort of present a near live stream of images) and append to a html image tag or just in the body of the html page.

incomplete sample code: (mostly acquired from official sample or just codes from)

index.js

```
// basic variables
var app = require('express')();
var http = require('http').Server(app);
var io = require('socket.io')(http);

var fs = require('fs'); // required for file serving

http.listen(3000, function(){
  console.log('listening on *:3000');
});

// location to index.html
app.get('/', function(req, res){
  res.sendFile(__dirname + '/index.html');
});

// only to test chat sample code from sample
io.on('connection', function(socket){

  console.log('a user connected');
  // broadcast a message
  socket.broadcast.emit('chat message', 'System Broadcast Message: a user has been connected');
  socket.on('chat message', function(msg){
    io.emit('chat message', msg);
  });

  // trying to serve the image file from the server
  io.on('connection', function(socket){
    fs.readFile(__dirname + '/images/image.jpg', function(err, buf){
      // it's possible to embed binary data
      // within arbitrarily-complex objects
      socket.emit('image', { image: true, buffer: buf });
      console.log('image file is initialized');
    });
  });
});
```

(client side html page) index.html (we'll cut to the chase with only the portion which serves the image) What can we do on the client side to get the file and serve the image on the html page?

```
socket.on("image", function(image, buffer) {
  if(image)
  {
    console.log(" image: from client side");
    // code to handle buffer like drawing with canvas** <--- is canvas drawing/library a requirement?
    is there an alternative? another quick and dirty solution?
    console.log(image);
    // what can we do here to serve the image onto an img tag?
  }

});
```

thank you for reading

Update:

after the code snippets from below it also needed to change "buffer" variable to image.buffer in order for the image to display correctly

basically change the line from

```
img.src = 'data:image/jpeg;base64,' + buffer;
```

To

UK Citizen Can Make £950 Daily - Brits Investing Secret Exposed

This Investment Secret Can Change UK Residents Life Forever! Brits Must Try This

Ad

trendizia.com

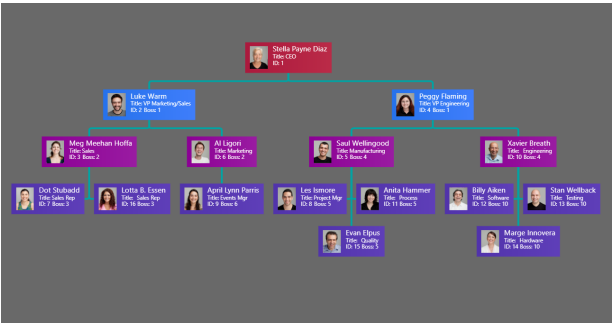
Visit Site

Share

Twitter

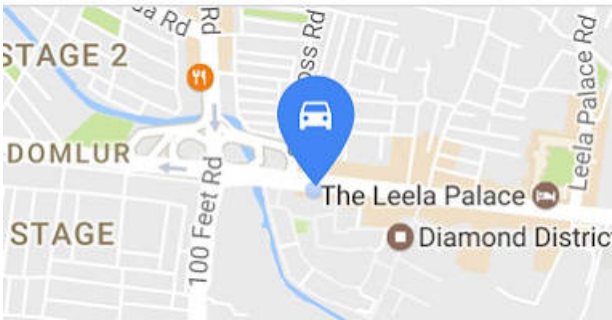
Facebook

Tags



JavaScript Graphics

Ad gojs.net



google-maps-markers
change position - Custom
marker in google maps...

i-harness.com

Custom URL Scheme	
Opening instagram:// , followed by one of the following parameters, will open our app and perform a custom action. For example, for camera, you would direct users on the iPhone to the custom URL instagram://camera .	
URL	OPENS
app	The Instagram app
camera	The camera (or photo library, on non-camera devices)
mediaId=MEDIA_ID	Media with this ID
userId=USERNAME	User with this username
locationId=LOCATION_ID	Location feed for this location ID
tagName=TAG	Tag feed for this tag

objective-c swift from -
How to share an image on
Instagram in iOS

i-harness.com



firebase swift - How do I
perform a pagination
query on Firebase's iOS...

i-harness.com



Born Before 2001?

Ad quotesearch.com



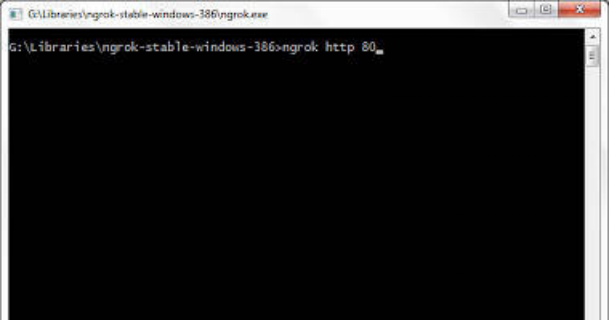
video convert to -
Streaming via RTSP or
RTP in HTML5

i-harness.com



initializer element - static
NSDictionary* const
letterValues = @{ ... } in...

i-harness.com



ip-address allow
accessing - Access
XAMPP Localhost from...

i-harness.com