

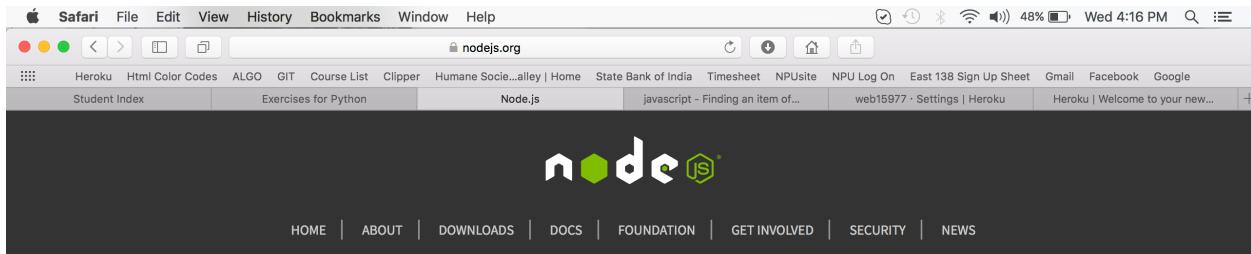
Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com

Implementation of WebSocket Server on node.js

Website Link: web15977.herokuapp.com

Installation of Node.js :



Node.js® is a JavaScript runtime built on [Chrome's V8 JavaScript engine](#). Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js' package ecosystem, [npm](#), is the largest ecosystem of open source libraries in the world.

Important security releases, please update now!

Download for OS X (x64)

v4.3.1 LTS

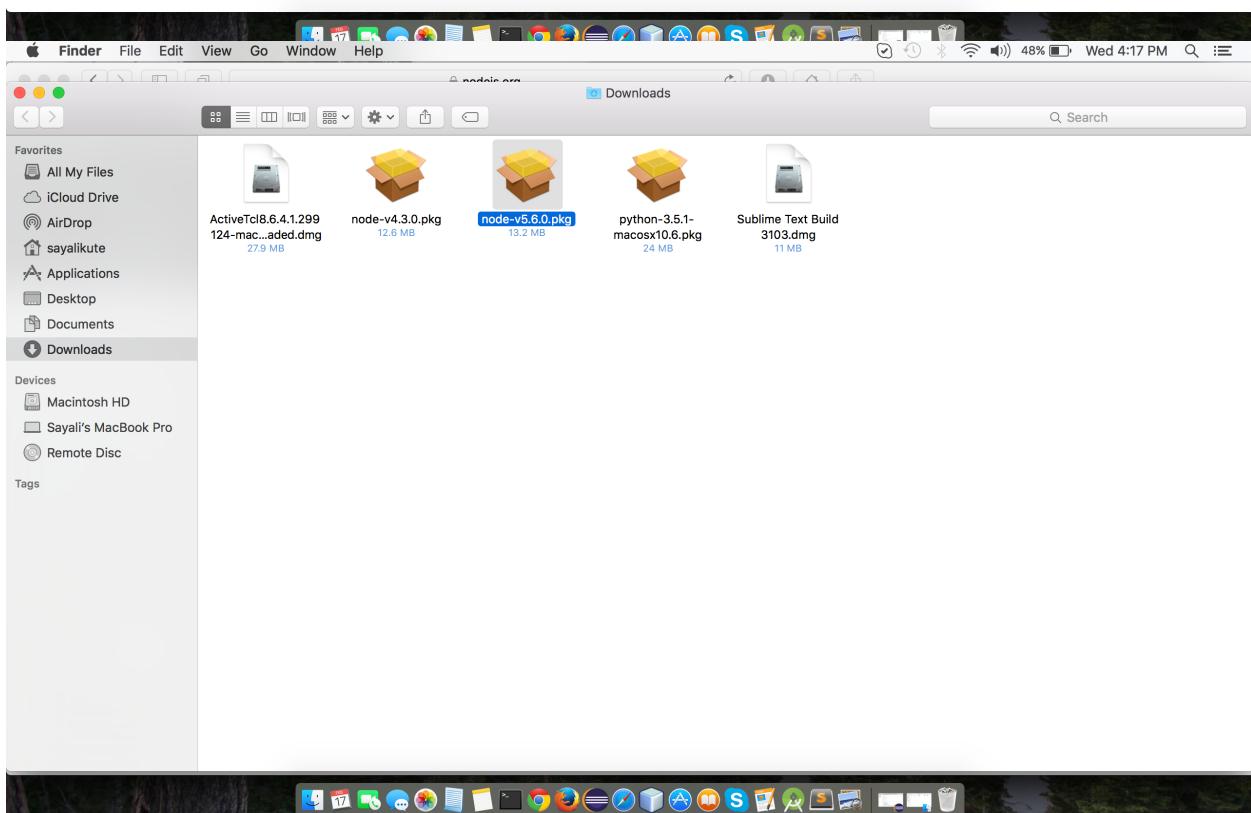
Mature and Dependable

v5.6.0 Stable

Latest Features

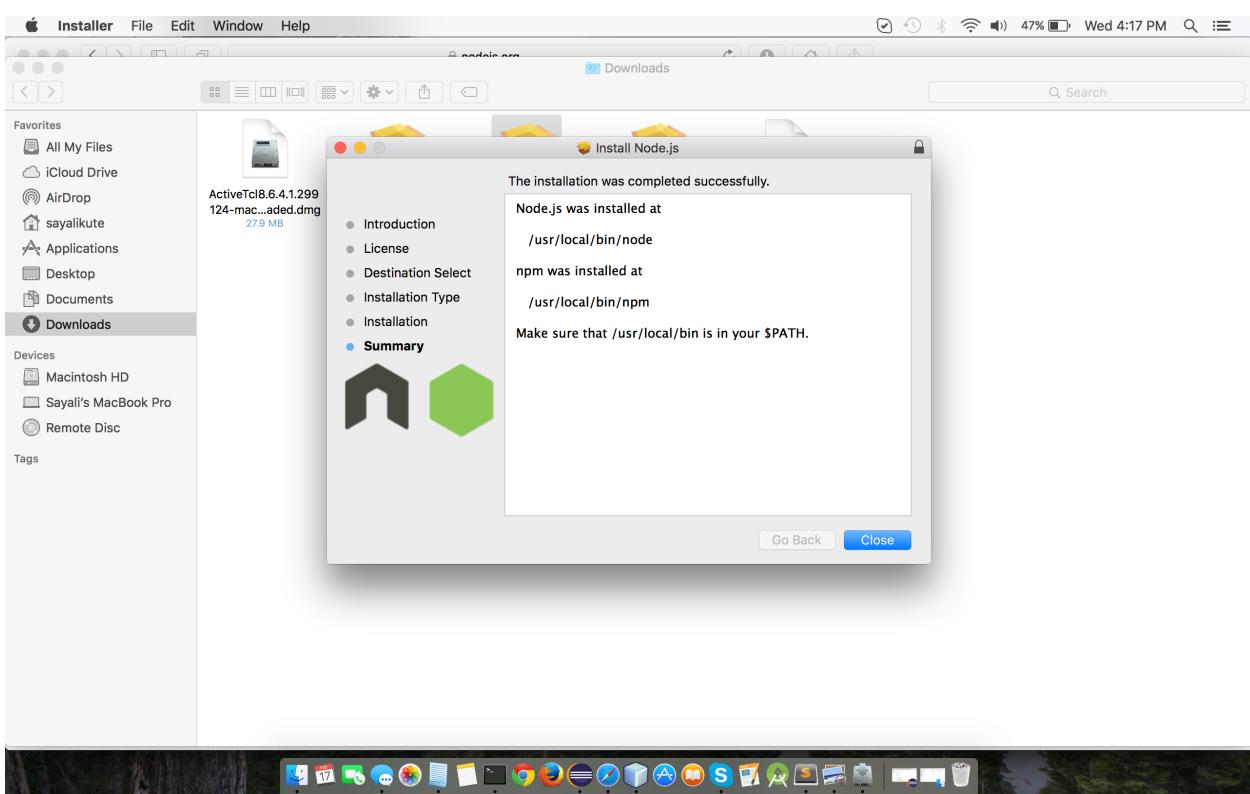
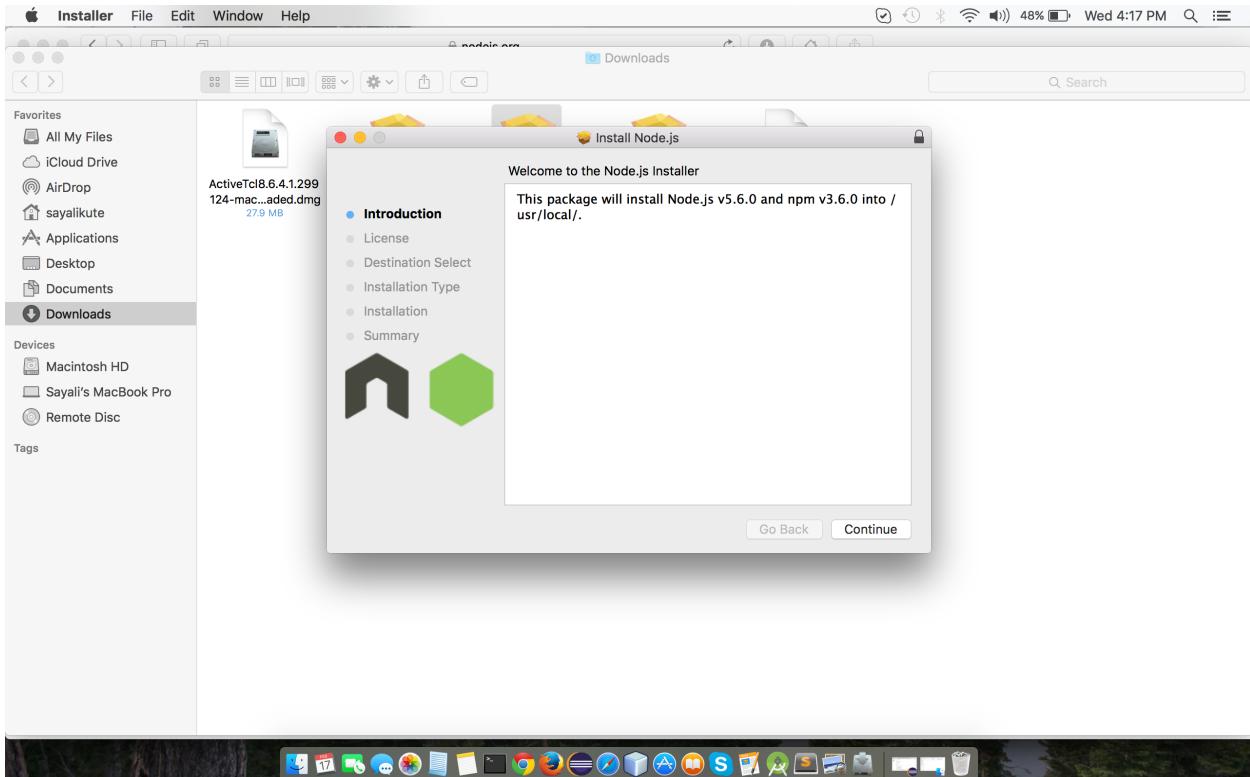
[Other Downloads](#) | [Changelog](#) | [API Docs](#) [Other Downloads](#) | [Changelog](#) | [API Docs](#)

Or have a look at the [LTS schedule](#).



Name: Sayali Vishnu Kute

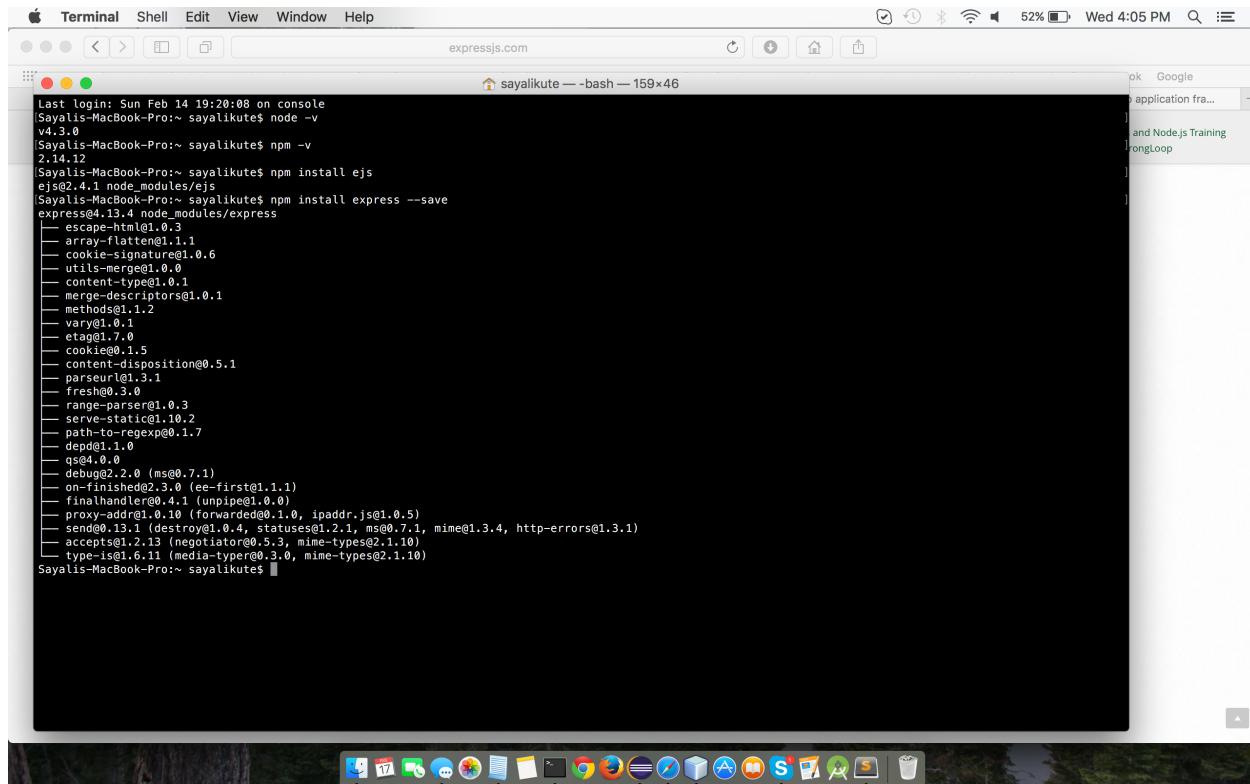
Email ID: sayalikute2507@gmail.com



Name: Sayali Vishnu Kute

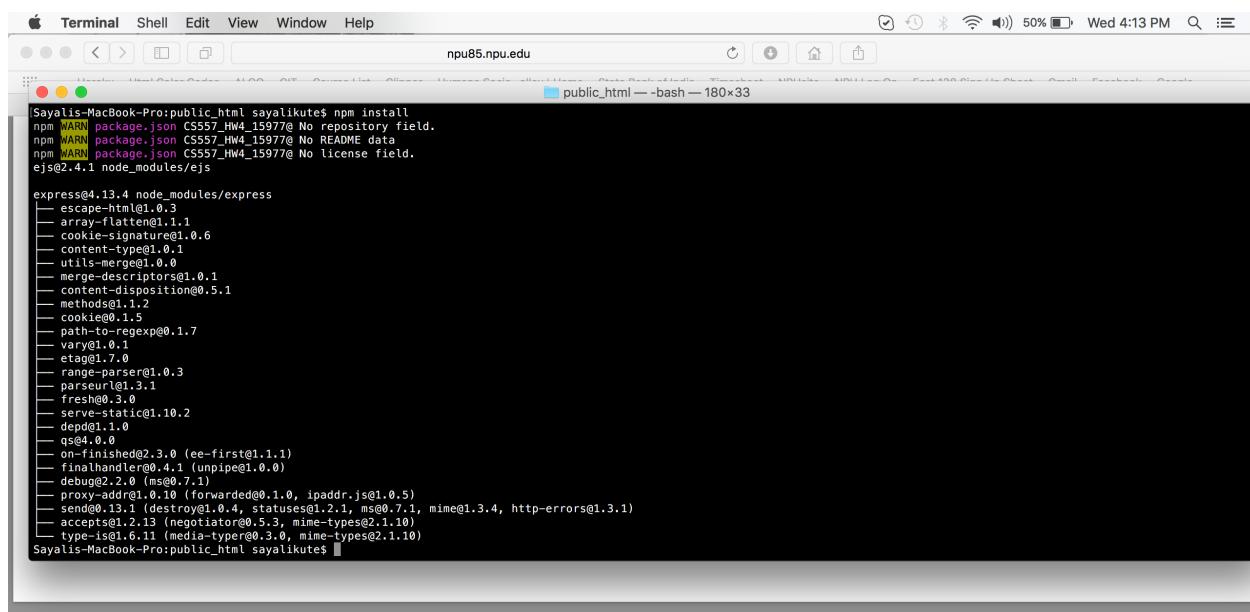
Email ID: sayalikute2507@gmail.com

Check node version and NPM version:



```
Last login: Sun Feb 14 19:20:00 on console
[Sayalis-MacBook-Pro:~ sayalikute]$ node -v
v4.3.0
[Sayalis-MacBook-Pro:~ sayalikute]$ npm -v
2.4.1
[Sayalis-MacBook-Pro:~ sayalikute]$ npm install ej
ej@2.4.1 node_modules/ej
[Sayalis-MacBook-Pro:~ sayalikute]$ npm install express --save
express@4.13.4 node_modules/express
  escape-html@0.1.3
  array-flatten@1.1.1
  cookie-signature@1.0.6
  utils-merge@1.0.0
  content-type@1.0.1
  merge-descriptors@1.0.1
  methods@1.1.2
  vary@1.0.1
  etag@1.7.0
  cookie@0.1.5
  content-disposition@0.5.1
  parseurl@1.3.1
  fresh@0.3.0
  range-parser@1.0.3
  serve-static@1.10.2
  path-to-regexp@0.1.7
  depd@1.1.0
  qs@4.0.0
  debug@2.2.0 (ms@0.7.1)
  on-finished@2.3.0 (ee-first@1.1.1)
  finalhandler@0.4.1 (unpipe@1.0.0)
  proxy-addr@1.0.10 (forwarded@0.1.0, ipaddr.js@1.0.5)
  send@0.13.1 (destroy@1.0.4, statuses@1.2.1, ms@0.7.1, mime@1.3.4, http-errors@1.3.1)
  accepts@1.2.13 (negotiator@0.5.3, mime-types@2.1.10)
  type-is@1.6.11 (media-types@0.3.0, mime-types@2.1.10)
[Sayalis-MacBook-Pro:~ sayalikute]$
```

Install Node Package Manager (NPM):



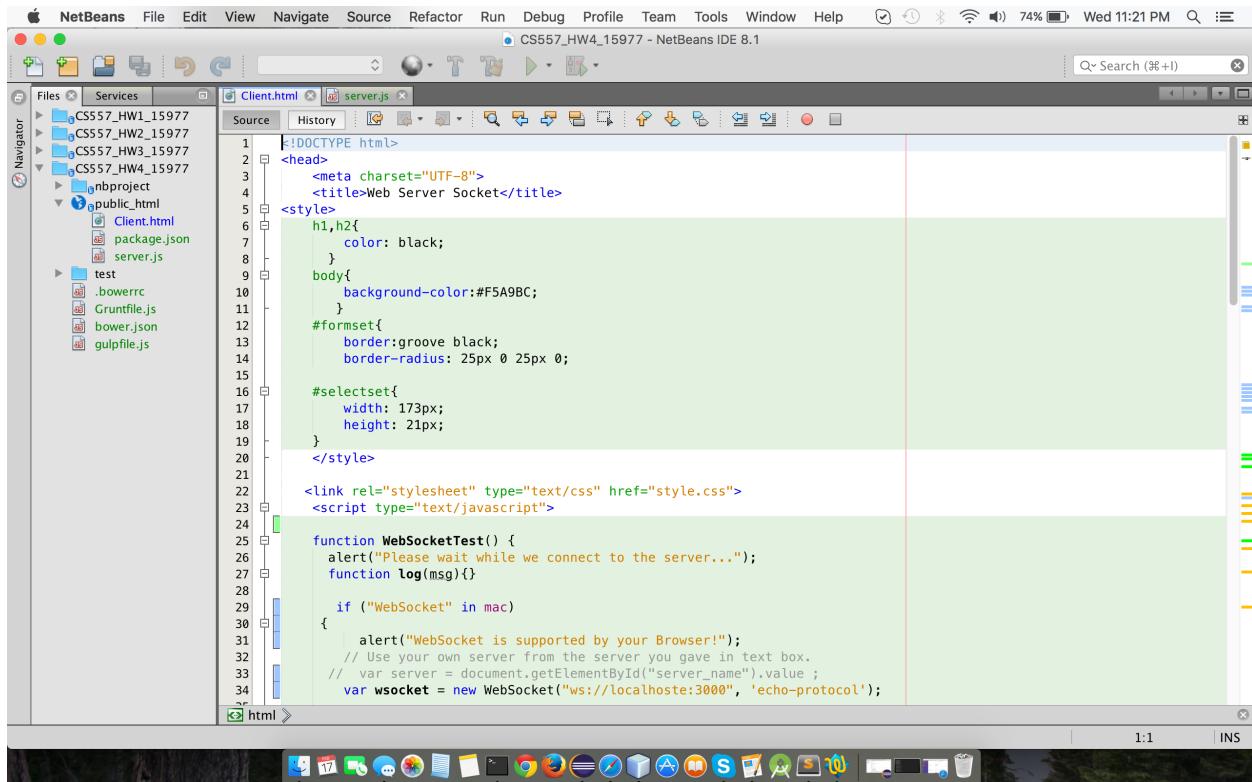
```
npm WARN package.json CS557_HW4_15977@ No repository field.
npm WARN package.json CS557_HW4_15977@ No README data
npm WARN package.json CS557_HW4_15977@ No license field.
ejs@2.4.1 node_modules/ejs

express@4.13.4 node_modules/express
  escape-html@0.1.3
  array-flatten@1.1.1
  cookie-signature@1.0.6
  content-type@1.0.1
  utils-merge@1.0.0
  merge-descriptors@1.0.1
  content-disposition@0.5.1
  methods@1.1.2
  cookie@0.1.5
  path-to-regexp@0.1.7
  depd@1.1.0
  qs@4.0.0
  on-finished@2.3.0 (ee-first@1.1.1)
  finalhandler@0.4.1 (unpipe@1.0.0)
  proxy-addr@1.0.10 (forwarded@0.1.0, ipaddr.js@1.0.5)
  send@0.13.1 (destroy@1.0.4, statuses@1.2.1, ms@0.7.1, mime@1.3.4, http-errors@1.3.1)
  accepts@1.2.13 (negotiator@0.5.3, mime-types@2.1.10)
  type-is@1.6.11 (media-types@0.3.0, mime-types@2.1.10)
[Sayalis-MacBook-Pro:public_html sayalikute]$
```

Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com

Create Client.html file as follows:



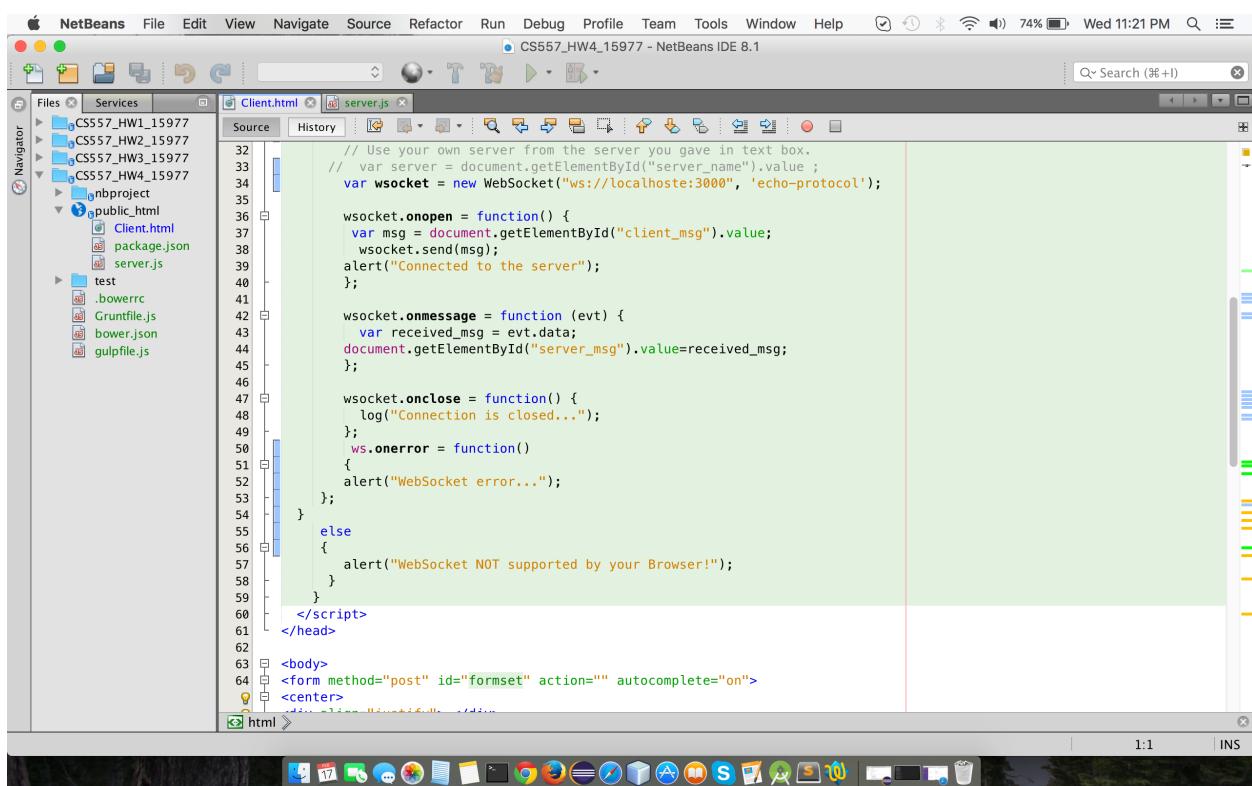
The screenshot shows the NetBeans IDE interface with the Client.html file open in the editor. The code is as follows:

```
<!DOCTYPE html>
<head>
    <meta charset="UTF-8">
    <title>Web Server Socket</title>
    <style>
        h1,h2{
            color: black;
        }
        body{
            background-color:#F5A9BC;
        }
        #formset{
            border:groove black;
            border-radius: 25px 0 25px 0;
        }
        #selectset{
            width: 173px;
            height: 21px;
        }
    </style>
    <link rel="stylesheet" type="text/css" href="style.css">
    <script type="text/javascript">

        function WebSocketTest() {
            alert("Please wait while we connect to the server...");
            function log(msg){}

            if ("WebSocket" in mac)
            {
                alert("WebSocket is supported by your Browser!");
                // Use your own server from the server you gave in text box.
                // var server = document.getElementById("server_name").value ;
                var wssocket = new WebSocket("ws://localhost:3000", 'echo-protocol');
            }
        }

    </script>
</head>
```



The screenshot shows the NetBeans IDE interface with the Client.html file open in the editor. The code has been expanded to include the JavaScript logic for connecting to the WebSocket server:

```
// Use your own server from the server you gave in text box.
// var server = document.getElementById("server_name").value ;
var wssocket = new WebSocket("ws://localhost:3000", 'echo-protocol');

wssocket.onopen = function() {
    var msg = document.getElementById("client_msg").value;
    wssocket.send(msg);
    alert("Connected to the server");
};

wssocket.onmessage = function (evt) {
    var received_msg = evt.data;
    document.getElementById("server_msg").value=received_msg;
};

wssocket.onclose = function() {
    log("Connection is closed...");
};
ws.onerror = function()
{
    alert("WebSocket error...");
};

else
{
    alert("WebSocket NOT supported by your Browser!");
}
</script>
</head>

<body>
<form method="post" id="formset" action="" autocomplete="on">
<center>
```

Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com

The screenshot shows the NetBeans IDE interface. The left sidebar displays a project structure with files like Client.html, server.js, .bowerrc, Gruntfile.js, bower.json, and gulpfile.js. The right pane contains two tabs: 'Client.html' and 'server.js'. The 'Client.html' tab shows an HTML file with code for a web socket server. The 'server.js' tab shows a JavaScript file for a Node.js application using Express and Websocket.

```
<h2>Web Socket Server </h2>
<div id="content_wrapper"></div>
<div id="content">
<div>
<div><br/>
    Server Name: <input class="input_field" id="server_name" type="text" value="ws://web15977.herokuapp.com/" style="width:300px" /> <br><br>
    Client Message: <input class="input_field" id="client_msg" type="text" value="Hello, How are you?" />
    Server Message: <input class="input_field" id="server_msg" type="text" />
</div><br />
<table style="clear:both">
<tr>
<td width="100px">
        <input class="button" type="button" onclick="javascript:WebSocketTest()" value="Connect & Send" />
    </td>
    <td>
        <input class="button" type="button" onclick="disconnect();" value="Stop" />
    </td>
</tr>
</table>
</div>
<div class="cleaner"></div>
</center>
</form>
</body>
</html>
```

```
var express = require('express');
var http = require('http');
var ws = require('ws').server;
var app = express();

var port = process.env.PORT || 3000;
app.set('view engine', 'ejs');

var server = http.createServer(app);
server.listen(port);

function log(msg){
  console.log((new Date()) + ' ' + msg);
}

server.on('listening', function(){
  log("Listening port " + port);
  log("In your browser open http://127.0.0.1:" + port);
});

var wsServer = new ws.Server({
  httpServer: server,
  autoAcceptConnections: false
});
wsServer.on('request', function(request) {
  var connection = request.accept('echo-protocol', request.origin);
  log('Connection accepted.');
  connection.on('message', function(message) {
    if (message.type === 'utf8') {
      log('Received Message: ' + message.utf8Data);
      connection.sendUTF("Hello, I'm fine, Thank you");
    }
  });
  connection.on('close', function(reasonCode, description) {
    log('Peer ' + connection.remoteAddress + ' disconnected.');
  });
});

app.get('/',function(req,res)
{
  res.sendFile(__dirname+'/Client.html');
});
```

Create server.js file as follows:

The screenshot shows a Notepad window with the 'server.js' file open. The code is identical to the one shown in the NetBeans screenshot, defining a Node.js application that serves an HTML page and handles web socket connections.

```
var express = require('express');
var http = require('http');
var ws = require('ws').server;
var app = express();

var port = process.env.PORT || 3000;
app.set('view engine', 'ejs');

var server = http.createServer(app);
server.listen(port);

function log(msg){
  console.log((new Date()) + ' ' + msg);
}

server.on('listening', function(){
  log("Listening port " + port);
  log("In your browser open http://127.0.0.1:" + port);
});

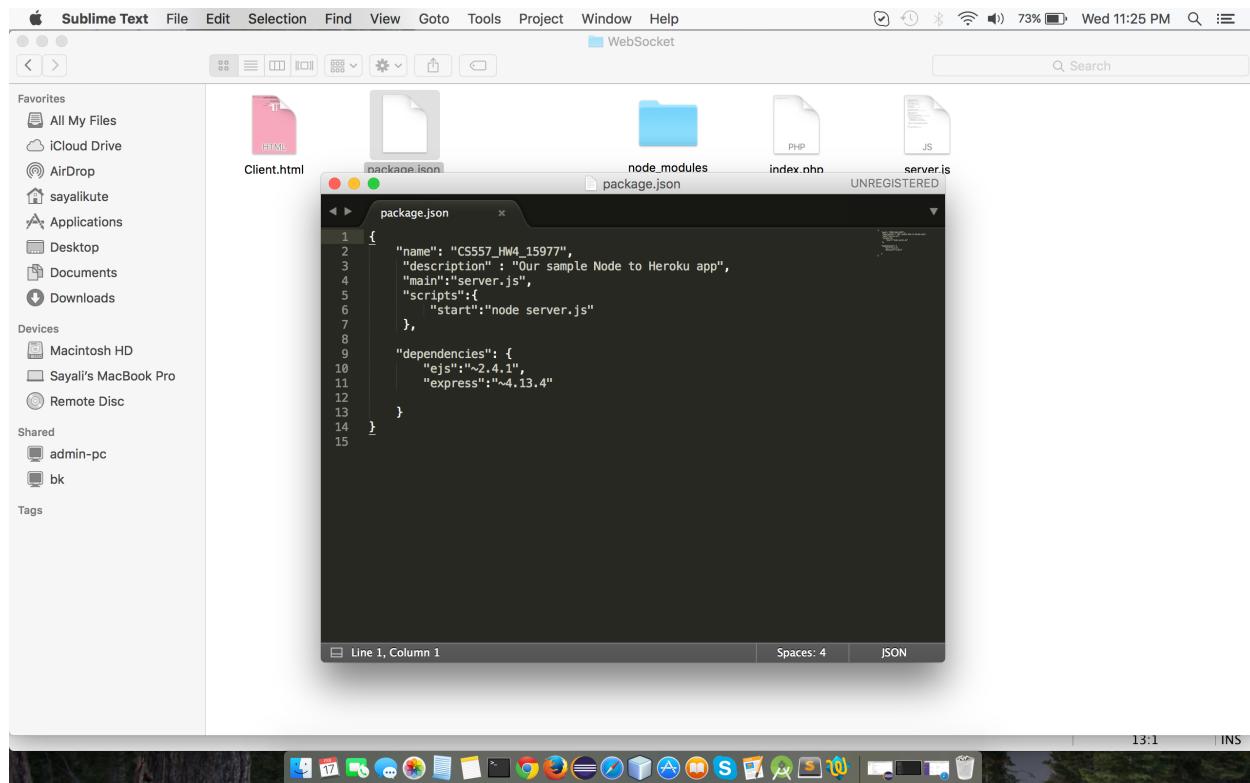
var wsServer = new ws.Server({
  httpServer: server,
  autoAcceptConnections: false
});
wsServer.on('request', function(request) {
  var connection = request.accept('echo-protocol', request.origin);
  log('Connection accepted.');
  connection.on('message', function(message) {
    if (message.type === 'utf8') {
      log('Received Message: ' + message.utf8Data);
      connection.sendUTF("Hello, I'm fine, Thank you");
    }
  });
  connection.on('close', function(reasonCode, description) {
    log('Peer ' + connection.remoteAddress + ' disconnected.');
  });
});

app.get('/',function(req,res)
{
  res.sendFile(__dirname+'/Client.html');
});
```

Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com

Create package.json file:



The screenshot shows a Sublime Text window displaying a package.json file. The file contains the following JSON code:

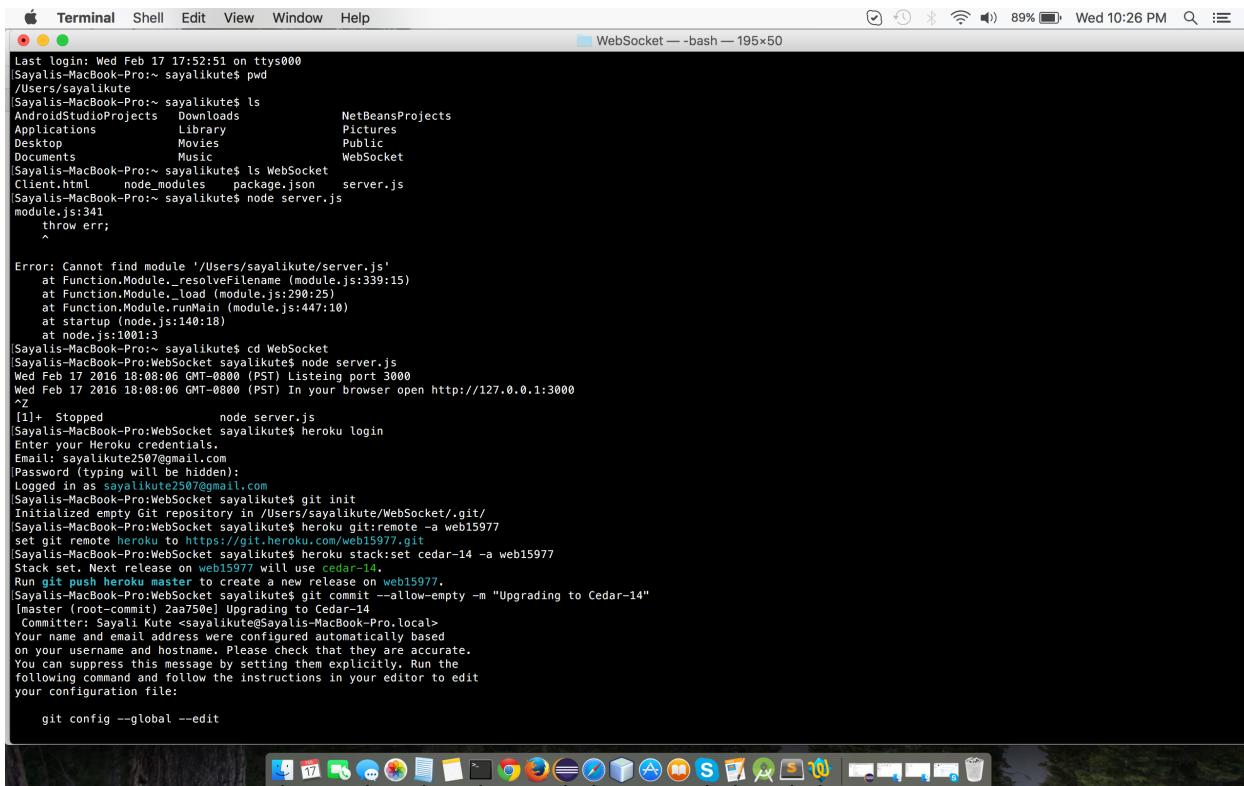
```
1 {  
2   "name": "CS557_HW4_15977",  
3   "description": "Our sample Node to Heroku app",  
4   "main": "server.js",  
5   "scripts": {  
6     "start": "node server.js"  
7   },  
8   "dependencies": {  
9     "ejs": "~2.4.1",  
10    "express": "~4.13.4"  
11  }  
12 }  
13 }  
14 }  
15 }
```

The Sublime Text interface includes a sidebar with file navigation, a toolbar with various icons, and a status bar at the bottom showing the time as 13:11.

Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com

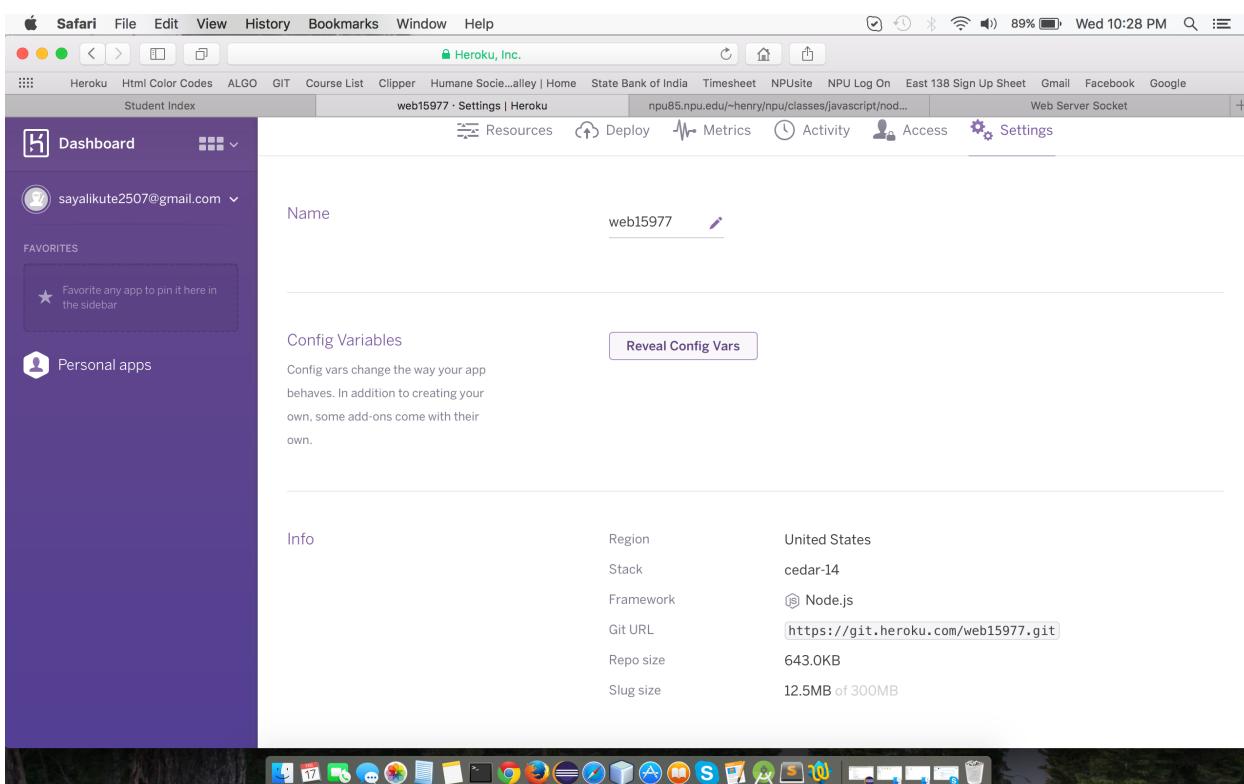
Running server.js on local host on port 3000 and further perform the Heroku login steps:



```
Last login: Wed Feb 17 17:52:51 on ttys000
[Sayalis-MacBook-Pro:~ sayalikute$ pwd
/Users/sayalikute
[Sayalis-MacBook-Pro:~ sayalikute$ ls
AndroidStudioProjects Downloads NetBeansProjects
Applications Library Pictures
Desktop Movies Public
Documents Music WebSocket
[Sayalis-MacBook-Pro:~ sayalikute$ ls WebSocket
Client.html node_modules package.json server.js
[Sayalis-MacBook-Pro:~ sayalikute$ node server.js
module.js:341
    throw err;
^

Error: Cannot find module '/Users/sayalikute/server.js'
  at Function.Module._resolveFilename (module.js:339:15)
  at Function.Module._load (module.js:290:25)
  at Function.Module.runMain (module.js:447:10)
  at startup (node.js:148:18)
  at node.js:1001:3
[Sayalis-MacBook-Pro:~ sayalikute$ cd WebSocket
[Sayalis-MacBook-Pro:WebSocket sayalikute$ node server.js
Wed Feb 17 2016 18:08:06 GMT-0800 (PST) Listening port 3000
Wed Feb 17 2016 18:08:06 GMT-0800 (PST) In your browser open http://127.0.0.1:3000
~
[1]+  Stopped                  node server.js
[Sayalis-MacBook-Pro:WebSocket sayalikute$ heroku login
Enter your Heroku credentials.
Email: sayalikute2507@gmail.com
>Password (typing will be hidden):
Logged in as sayalikute2507@gmail.com
[Sayalis-MacBook-Pro:WebSocket sayalikute$ git init
Initialized empty Git repository in /Users/sayalikute/WebSocket/.git/
[Sayalis-MacBook-Pro:WebSocket sayalikute$ heroku git:remote -a web15977
set git remote heroku to https://git.heroku.com/web15977.git
[Sayalis-MacBook-Pro:WebSocket sayalikute$ heroku stack:set cedar-14 -a web15977
Stack set. Next release on web15977 will use cedar-14.
Run git push heroku master to create a new release on web15977.
[Sayalis-MacBook-Pro:WebSocket sayalikute$ git commit --allow-empty -m "Upgrading to Cedar-14"
[master (root-commit) 2aa750e] Upgrading to Cedar-14
Committer: Sayali Kute <sayalikute@Sayali-MacBook-Pro.local>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
git config --global --edit
```

Create an app on the Heroku website:

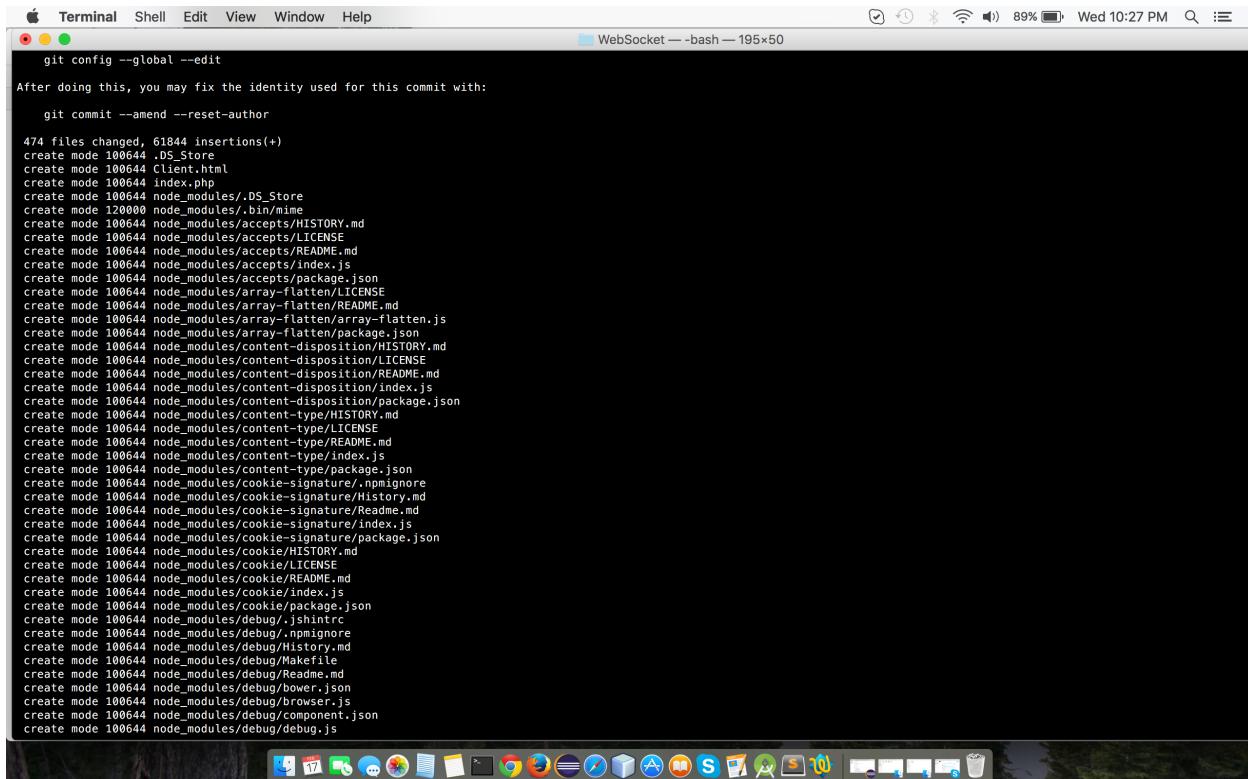


The screenshot shows the Heroku dashboard for an application named 'web15977'. The dashboard includes sections for Resources, Deploy, Metrics, Activity, Access, and Settings. Under the 'Info' section, the app's details are listed as follows:

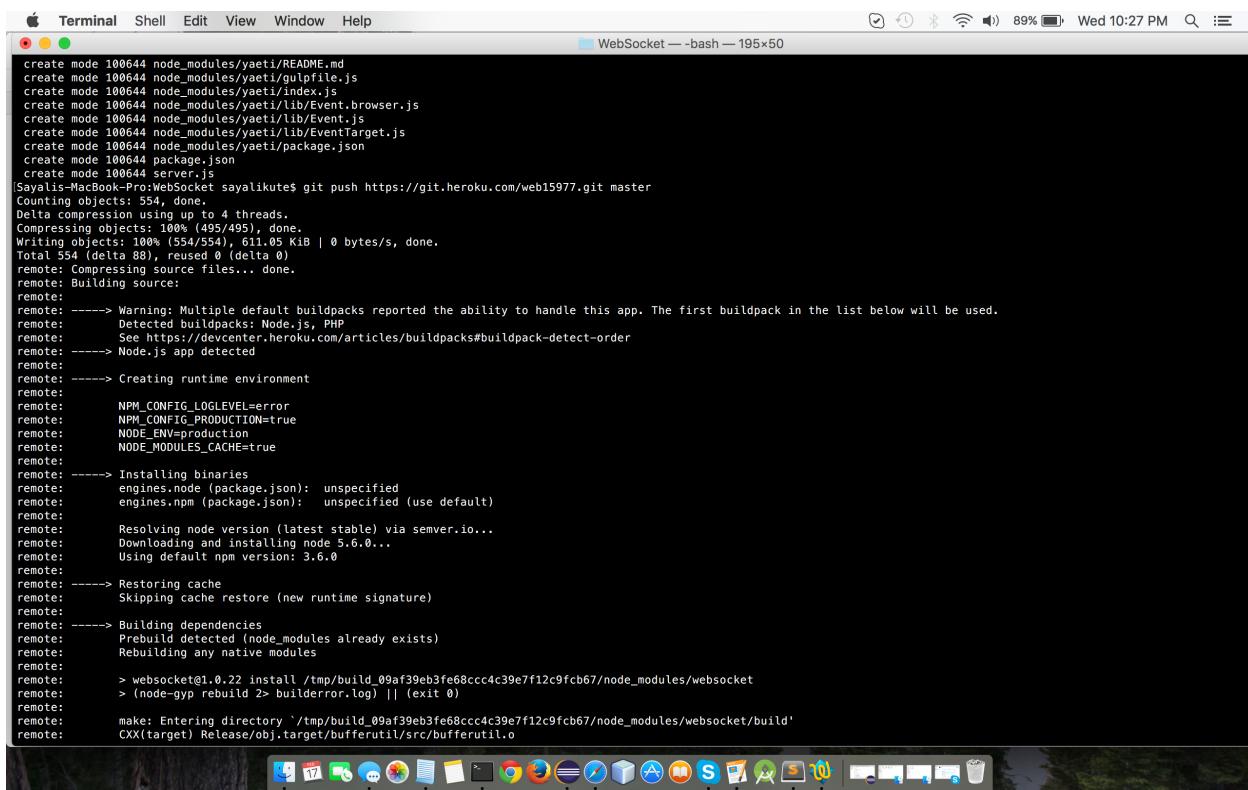
Info	Value
Region	United States
Stack	cedar-14
Framework	Node.js
Git URL	https://git.heroku.com/web15977.git
Repo size	643.0KB
Slug size	12.5MB of 300MB

Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com



```
Terminal Shell Edit View Window Help
WebSocket — bash — 195x50
git config --global --edit
After doing this, you may fix the identity used for this commit with:
git commit --amend --reset-author
474 files changed, 61844 insertions(+)
create mode 100644 .DS_Store
create mode 100644 Client.html
create mode 100644 index.php
create mode 100644 node_modules/.DS_Store
create mode 120000 node_modules/.bin/nime
create mode 100644 node_modules/accepts/HISTORY.md
create mode 100644 node_modules/accepts/LICENSE
create mode 100644 node_modules/accepts/README.md
create mode 100644 node_modules/accepts/index.js
create mode 100644 node_modules/accepts/package.json
create mode 100644 node_modules/array-flatten/LICENSE
create mode 100644 node_modules/array-flatten/README.md
create mode 100644 node_modules/array-flatten/array-flatten.js
create mode 100644 node_modules/array-flatten/package.json
create mode 100644 node_modules/content-disposition/HISTORY.md
create mode 100644 node_modules/content-disposition/LICENSE
create mode 100644 node_modules/content-disposition/README.md
create mode 100644 node_modules/content-disposition/index.js
create mode 100644 node_modules/content-disposition/package.json
create mode 100644 node_modules/content-type/HISTORY.md
create mode 100644 node_modules/content-type/LICENSE
create mode 100644 node_modules/content-type/README.md
create mode 100644 node_modules/content-type/index.js
create mode 100644 node_modules/content-type/package.json
create mode 100644 node_modules/cookie-signature/.npmignore
create mode 100644 node_modules/cookie-signature/HISTORY.md
create mode 100644 node_modules/cookie-signature/Readme.md
create mode 100644 node_modules/cookie-signature/index.js
create mode 100644 node_modules/cookie-signature/package.json
create mode 100644 node_modules/cookie/HISTORY.md
create mode 100644 node_modules/cookie/LICENSE
create mode 100644 node_modules/cookie/README.md
create mode 100644 node_modules/cookie/index.js
create mode 100644 node_modules/cookie/package.json
create mode 100644 node_modules/debug/.jshintrc
create mode 100644 node_modules/debug/.npmignore
create mode 100644 node_modules/debug/HISTORY.md
create mode 100644 node_modules/debug/Makefile
create mode 100644 node_modules/debug/Readme.md
create mode 100644 node_modules/debug/bower.json
create mode 100644 node_modules/debug/browser.js
create mode 100644 node_modules/debug/component.json
create mode 100644 node_modules/debug/debug.js
```



```
Terminal Shell Edit View Window Help
WebSocket — bash — 195x50
create mode 100644 node_modules/yaeli/README.md
create mode 100644 node_modules/yaeli/gulpfile.js
create mode 100644 node_modules/yaeli/index.js
create mode 100644 node_modules/yaeli/lib/Event.browser.js
create mode 100644 node_modules/yaeli/lib/Event.js
create mode 100644 node_modules/yaeli/lib/EventTarget.js
create mode 100644 node_modules/yaeli/package.json
create mode 100644 package.json
create mode 100644 server.js
Sayalis-MacBook-Pro:WebSocket sayalikute$ git push https://git.heroku.com/web15977.git master
Counting objects: 554, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (495/495), done.
Writing objects: 100% (554/554), 611.05 KiB | 0 bytes/s, done.
Total 554 (delta 88), reused 0 (delta 0)
remote: Compressing source files... done.
remote: Building source:
remote:
remote: -----> Warning: Multiple default buildpacks reported the ability to handle this app. The first buildpack in the list below will be used.
remote: Detected buildpacks: Node.js, PHP
remote: See https://devcenter.heroku.com/articles/buildpacks#buildpack-detect-order
remote: -----> Node.js app detected
remote:
remote: -----> Creating runtime environment
remote:
remote:   NPM_CONFIG_LOGLEVEL=error
remote:   NPM_CONFIG_PRODUCTION=true
remote:   NODE_ENV=production
remote:   NODE_MODULES_CACHE=true
remote:
remote: -----> Installing binaries
remote:   engines.node (package.json): unspecified
remote:   engines.npm (package.json): unspecified (use default)
remote:
remote:     Resolving node version (latest stable) via semver.io...
remote:     Downloading and installing node 5.6.0...
remote:     Using default npm version: 3.6.0
remote:
remote: -----> Restoring cache
remote:   Skipping cache restore (new runtime signature)
remote:
remote: -----> Building dependencies
remote:   Prebuild detected (node_modules already exists)
remote:   Rebuilding any native modules
remote:
remote:     > websocket@1.0.22 install /tmp/build_09af39eb3fe68ccc4c39e7f12c9fc67/node_modules/websocket
remote:     > (node-gyp rebuild 2>> buildererror.log) || (exit 0)
remote:
remote:     make: Entering directory '/tmp/build_09af39eb3fe68ccc4c39e7f12c9fc67/node_modules/websocket/build'
remote:       CXX(target) Release/obj.target/bufferutil/src/bufferutil.o
```

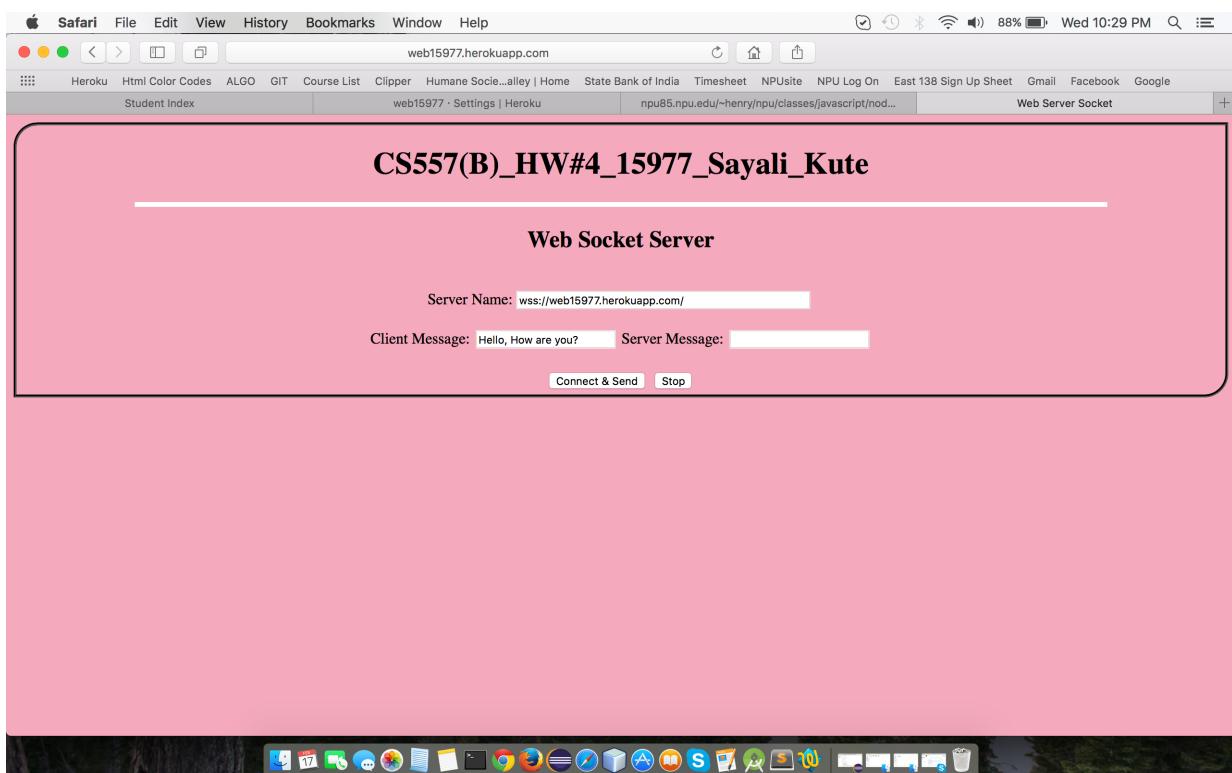
Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com

The folder is deployed on Heroku successfully:

```
Terminal Shell Edit View Window Help WebSocket — bash — 195x50
remote: ws@1.0.1: /tmp/build_09af39eb3fe68ccc4c39e7f12c9fc67/node_modules/ws
remote: options@0. 0.6 /tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/options
remote: ultron@1.0.2 /tmp/build_09af39eb3fe68ccc4c39e7f12c9fc67/node_modules/ultron
remote: is-typedarray@1.0.1 /tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/is-typedarray
remote: nan@0. 0.9 /tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/nan
remote: typedarray-to-buffer@0.3.0 /tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/typedarray-to-buffer
remote: websocket@0.2.2 /tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/websocket
remote: yaeti@0. 0.5 /tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/yaeti
remote: 
remote: Installing any new modules (package.json)
remote: 
remote: > websocket@1.0.22 install /tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/websocket
remote: > (node-gyp rebuild 2>> buildererror.log) || (exit 0)
remote: 
remote: make: Entering directory '/tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/websocket/build'
remote: CXX(target) Release/obj.target/bufferutil/src/bufferutil.o
remote: SOLINK_MODULE(target) Release/obj.target/bufferutil.node
remote: COPY Release/bufferutil.node
remote: CXX(target) Release/obj.target/validation/src/validation.o
remote: SOLINK_MODULE(target) Release/obj.target/validation.node
remote: COPY Release/validation.node
remote: make: Leaving directory '/tmp/build_09af39eb3fe68cc4c39e7f12c9fc67/node_modules/websocket/build'
remote: C5557_HW4_15977@ /tmp/build_09af39eb3fe68cc4c39e7f12c9fc67
remote: └─ websocket@1.0.22 extraneous
remote: 
remote: 
remote: -----> Caching build
remote: Clearing previous node cache
remote: Saving 2 cacheDirectories (default):
remote: - node_modules
remote: - bower_components (nothing to cache)
remote: 
remote: -----> Build succeeded!
remote:   ej@2.4.1
remote:   express@4.13.4
remote:   websocket@1.0.22 extraneous
remote:   ws@1.0.1
remote: 
remote: -----> Discovering process types
remote: Procfile declares types      -> (none)
remote: Default types for buildpack -> web
remote: 
remote: -----> Compressing...
remote: Done: 12.5M
remote: -----> Launching...
remote: Released v3
remote: https://web15977.herokuapp.com/ deployed to Heroku
remote: 
remote: Verifying deploy.... done.
To https://git.heroku.com/web15977.git
 * [new branch]      master -> master
```

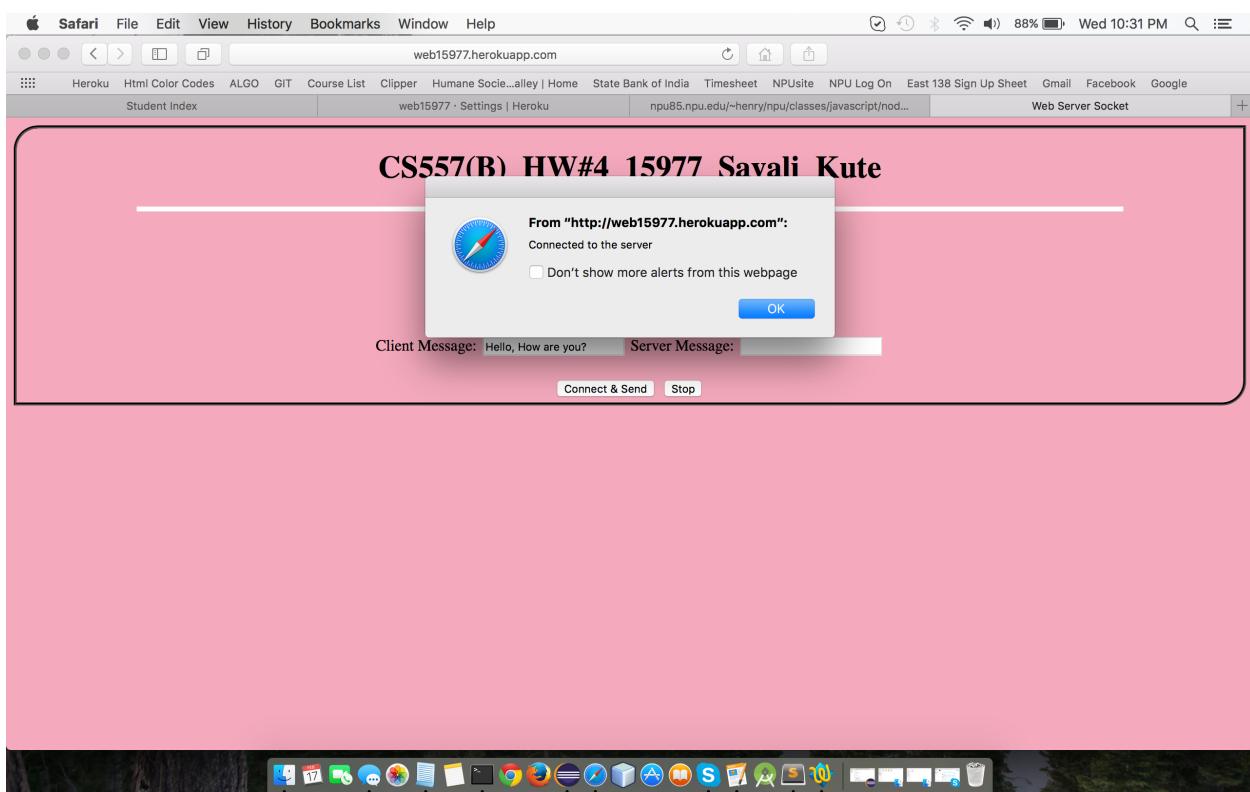
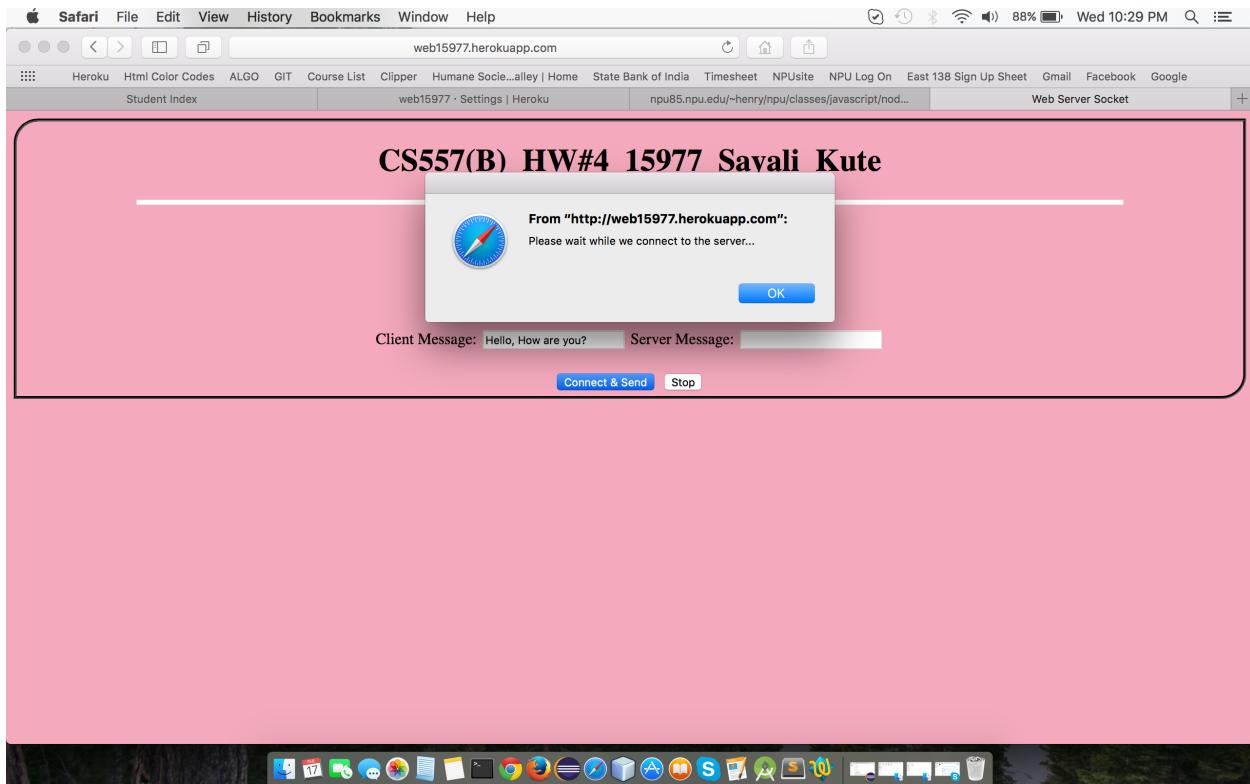
Open web15977.herokuapp.com on the browser:



Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com

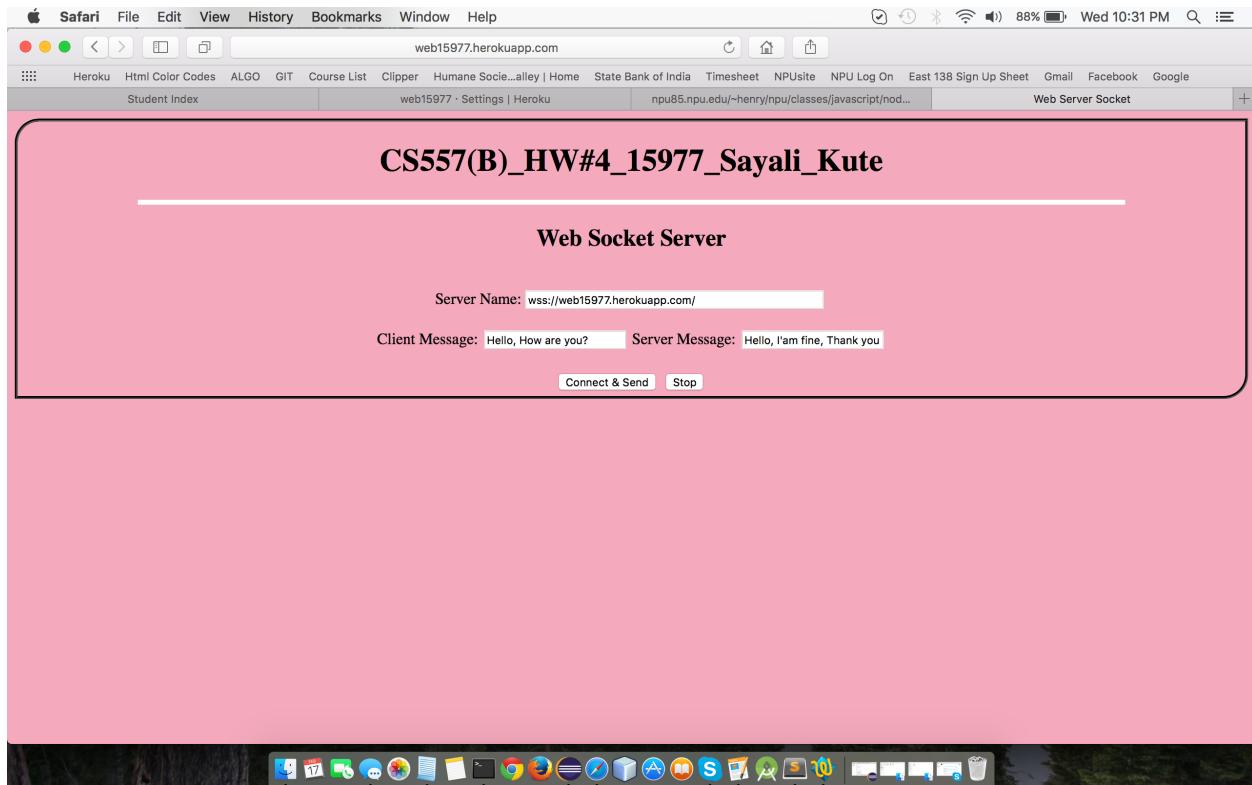
Client Send a message to server along with the connection request:



Name: Sayali Vishnu Kute

Email ID: sayalikute2507@gmail.com

A response message sent by the Server is seen as follows:



—————END—————