

Saturday, March 4, 2023 10:46 AM

app.component.ts:48

idToken can apparently fetch everything here. Its like the ultimate token. Just have that token and you have all this info



How To Implement Login With Google In Angular

- ▼ homepage
 - # homepage.component.css
 - 🔗 homepage.component.html
 - TS homepage.component.spec.ts
 - TS homepage.component.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-homepage',
  templateUrl: './homepage.component.html',
  styleUrls: ['./homepage.component.css']
})
export class HomepageComponent {
}
```

```
<title>AngularFortune</title>
<base href="/login">
```

However, what I learned is that it doesn't navigate to component `/login`. Its just a rename

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { LoginComponent } from '../views/login/login.component';
import { UserpageComponent } from '../views/userpage/userpage.component';
import { HomepageComponent } from '../views/homepage/homepage.component';

const routes: Routes = [
  { path: '', component: HomepageComponent }, //this is homepage
  { path: 'login', component: LoginComponent },
  { path: 'userpage', component: UserpageComponent }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})

export class AppRoutingModuleModule { }
//this will automatically import all the components we rout to. All we need to do is update this list
export const routingComponents = [ HomepageComponent, LoginComponent, UserpageComponent]
```

```
@NgModule({
  declarations: [
    AppComponent,
    routingComponents
  ],
```

Notice that the home

```
*
* - Trying to Integrate backend to front end, following the youtube tutorial below roughly
* - https://youtube.com/playlist?list=PLlameCF3cMEtXtMLOfV6V45ze1To0LTb
* - It incorporates fiber and pusher
* - I'm testing using postman
*/

package main
import (
    "github.com/gofiber/fiber/v2"
    "github.com/gofiber/fiber/v2/middleware/cors"
    "github.com/pusher/pusher-http-go/v5"
    "fmt"
)
```

```

func main() {
  app := fiber.New()

  app.Use(cors.New())

  pusherClient := pusher.Client{
    AppID: "1564867",
    Key: "af21a1a5218dda4b051a",
    Secret: "b73ef85082132de68896",
    Cluster: "us2",
    Secure: true,
  }

  app.Post("/api/messages", func(c *fiber.Ctx) error {
    var data map[string]string

    if err := c.BodyParser(&data); err != nil {
      return err
    }

    //The channel is 'chat', the event is 'message', the data we want to send is 'data'
    err := pusherClient.Trigger("chat", "message", data)
    if err != nil {
      fmt.Println(err.Error())
    }

    return c.JSON([]string{})
  })

  /* Front end and Back end runs on different ports
  //This needs to be the case for front end to request from backend.
  app.Listen(":8000")
}

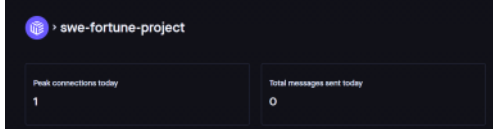
```

Make sure that the go program is running before you send any requests through postman

Then in postman:



Then notice before you send, in Pusher



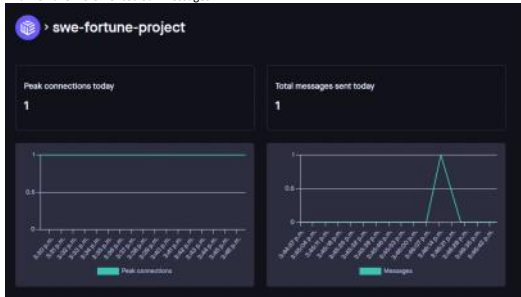
We're looking at the total messages sent today. Once we sent this request, the message should increment.

Now sending...

As expected, we get an empty array [] in response



And we have incremented our messages



Now for frontend, I'm starting by using bootstrap for the template and all that I did all the changes in homepage. Note that I had to import bootstrap and also include in appmodules:

```

import { FormsModule } from '@angular/forms';
Gotta do npm install pusher-js

```