

# Network Programming

## Lecture 7: Login

Mahmoud El-Gayyar

[elgayyar@ci.suez.edu.eg](mailto:elgayyar@ci.suez.edu.eg)

[Elgayyar.weebly.com](http://Elgayyar.weebly.com)



# Registration

1. ***Setup Satellizer for Login***
2. *Create a Login endpoint with Express*
3. *Show Logout in our navbar*

# Add login url to your config

- Configuration (*index.config.js*)

```
export function config ($logProvider, toastrConfig, $authProvider, API_URL) {  
  'ngInject' ;  
  
  ...  
  $authProvider.signupUrl = API_URL + 'auth/register';  
  
  $authProvider.loginUrl = API_URL + 'auth/login';  
  
}
```

# Bind Login Form to Satellizer

- Add *ng-submit* to the loginform (*auth.html*) and bind values to user model

```
<form name="login" ng-submit="auth.login()">
  ...
  <input type="email" ng-model="auth.login.user.email">
  <input typ="password" ng-model="auth.login.user.pwd">
```

- Create the login function (*auth.controller.js*)
  - ◆ Test in your browser and hit submit in register form. You should get **404 error!!**

```
login() {
  var vm=this;
  this.$auth.login(this.login.user).then(function(token) {
    vm.$auth.setToken(token);
  });
}
```

# Registration

1. *Setup Satellizer for Login*
2. ***Create a Login endpoint with Express***
3. *Show Logout in our navbar*

# Create a login endpoint with Express

- *Add new endpoint (server.js)*

```
app.post('/auth/login', auth.login);
```

- *Now add the login function to the controller*

# Add Login Function to Auth Controller

- *Update controller/auth.js*

```
module.exports = {
  register: function (req, res) {
    ...
  },
  login: function (req, res) {
    User.findOne({
      email: req.body.email
    }, function (err, user) {

      if (!user)
        return res.status(401).send({
          message: 'Email or Password invalid'
        });

      if (req.body.pwd == user.pwd) {
        console.log(req.body, user.pwd)
        res.send({
          token: createToken(user)
        });
      } else {
        return res.status(401).send({
          message: 'Invalid email and/or password'
        });
      }
    });
  }
};
```

# Registration

1. *Setup Satellizer for Login*
2. *Create a Login endpoint with Express*
3. ***Show Logout in our navbar***



# Add Navbar Controller

- We need to have logout button instead of the login button if we are already logged in.
- So we need to add code to the navbar to handle that
  - ◆ So you will need a controller
  - ◆ Create new file: app/components/navbar/**navbar.controller.js**

```
export class NavbarController {  
  
    constructor($auth) { //satellizer service  
        'ngInject';  
  
        this.$auth = $auth;  
    }  
  
}
```

- ◆ Register your controller inside **index.module.js**

```
import { MainController } from './main/main.controller';  
import { NavbarController } from './components/navbar/navbar.controller';  
...  
.controller('MainController', MainController)  
.controller('NavbarController', NavbarController)
```

# What about the Router

- *We don't need to add the controller to our router as it doesn't move us to another view (page)*
- *So we can use our controller directly in our html (navbar.html)*

```
<nav ng-controller="NavbarController as nav"  
      class="navbar navbar-static-top navbar-inverse">
```

# Get Authentication State

- Use *Satllizer isAuthenticated* method to get authentication state.

```
export class NavController {  
  
    constructor($auth) { //satellizer service  
        'ngInject';  
  
        this.$auth = $auth;  
        this.isAuthenticated = $auth.isAuthenticated;  
    }  
  
}
```

# Update Your Html to use the State

```
<ul class="nav navbar-nav">
  <li class="active"><a ng-href="#">Home</a></li>

  <li ng-hide="nav.isAuthenticated()" >
    <a ui-sref="auth">Login</a>
  </li>

  <li ng-show="nav.isAuthenticated()" >
    <a ng-click="nav.logout()" >Logout</a>
  </li>
</ul>
```

# Add the Logout function to the Controller

```
export class NavbarController {  
  
    constructor($auth) { //satellizer service  
        'ngInject';  
  
        this.$auth = $auth;  
        this.isAuthenticated = $auth.isAuthenticated;  
    }  
  
    logout() {  
        this.$auth.logout();  
    }  
  
}
```

Test Your App