

Workshop :

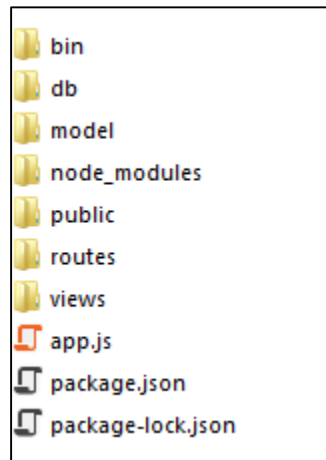
Communication entre deux applications Node JS et Angular

Objectifs :

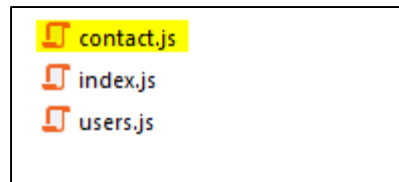
- Créer une application fullstack js en utilisant MongoDB, Node JS, Express JS et Angular
- Consommer des données d'une application node js dans une application Angular.

Travail à faire :

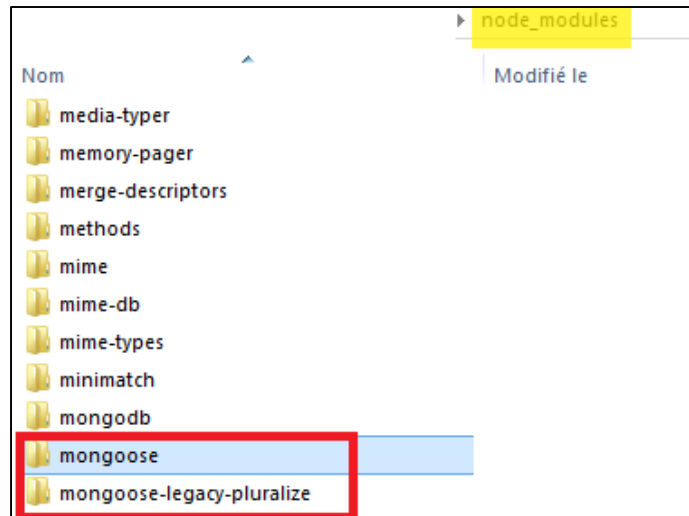
- 1- Créez un projet express nommé « **backend** » et installer le module « **Mongoose** ».



- 2- Dans le dossier « routes », Créer un contrôleur nommé « contact.js ».



- 3- Installer mongoose via la commande « **npm i mongoose** ».



- 4- Créer la collection « **Contact** » qui contient deux champs (**FullName** et **Phone**) et créer le schéma contact associé à cette collection.

```
var mongoose = require('mongoose');
var Schema = mongoose.Schema;

var Contact = new Schema({
  FullName : String,
  Phone : Number
});

module.exports = mongoose.model('contacts', Contact);
```

- 5- Dans le fichier « **routes/contact.js** » créer deux routes (get, post) permettant respectivement de récupérer les données de la base de données et d'insérer des données dans la base de données.

```
var Contact = require('../model/contact');
//Liste des contacts
router.get('/', function(req, res){
  Contact.find((err, data)=>{
    if(err) throw err;
    // res.render("showContact.twig", {data})
    res.json(data);
  });
});
```

```
//Ajouter Contact
router.post('/add', function(req, res){
  console.log(req.body);
  var c = new Contact({
    FullName : req.body.FullName,
    Phone : req.body.Phone
  });
  c.save();
});
```

Installer le module « cors » via la commande « npm i cors ».

```
PS E:\2021\chebbiMohamedAMine\CRUD_APP> npm i cors
```

Activer par la suite ce module dans le fichier app.js

```
var cors = require('cors');
app.use(cors());
```

NB : Le module cors permet de résoudre le problème [CORS](#).

Créer un projet Angular et en utilisant le module HttpClient faire la consommation des deux routes GET et POST.