How to make user registration with Node.js and MongoDB (using mongoose and Express.js)

I need your help. I want to make User registration form and use Nodejs, Express.js, MongoDB(mongoose) and give me very simple example how to make user registration form with: Name, Email, Password and Mobile Number :) I've made mongoose schema and give values like that Name: req.body.name but it won't work :/ In my oppinion I made something bad.

this is my code and if you think it's not correct, please correct it. (sorry for my bad english). this is server.js

```
var mongoose = require('mongoose');
mongoose.connect('mongodb://localhost/addressbookdb');
var express = require('express');
var app = express();
var db = mongoose.connection;

app.use(express.static(__dirname + '/../client'));

app.post("/",function(req,res){
    res.end("Registration Succesfully Completed!");

    var db = mongoose.connection;
    db.on('error', console.error.bind(console, 'connection error:'));
    db.once('open', function (callback) {
        console.log("connected.")
    });

    // Schema
    var RegSchema = mongoose.Schema({
        Name: String,
        Email: String,
        Pass: String,
        Num: Number,
        reg_time : {
            type : Date, default: Date.now
        }
    }, { collection: 'AddressCol' });

    // Model
    var UserReg = mongoose.model('UserReg', RegSchema);

    // Add in collection
    var UserAdd = new UserReg({
        Name: req.body.name,
        Email: req.body.email,
        Pass: req.body.pass,
        Num: req.body.num,
    });

    // Save
    UserAdd.save(function (err, fluffy) {
        if (err) return console.error(err);
    });
});

app.listen(8000, function() {
  console.log("Server is running!");
});
```

and this is my HTML page:

```
<div class="form-group">
    <input type="text" class="form-control" id="name" placeholder="name"><br>
```

```html
    <input type="email" class="form-control" id="email" placeholder="Email"><br>
    <input type="password" class="form-control" id="pass" placeholder="Password"><br>
    <input type="number" class="form-control" id="num" placeholder="Number"><br>
    <button type="submit" class="btn btn-primary" id="reg-form-btn">Registration!</button>
</div>
<script>
    $(document).ready(function() {
        $("#reg-form-btn").click(function() {
            var name = $("#name").val();
            var email = $("#email").val();
            var pass = $("#pass").val();
            var num = $("#num").val();
            $.post("/", {
                Name: name,
                Email: email,
                Pass: pass,
                Num: num
            });
        });
    });
</script>
```

javascript   node.js   mongodb   express   mongoose

The code is completely written in a wrong way. Please go though the tutorial and check how it's done.
scotch.io/tutorials/… – Krishnandu Sarkar Aug 13 '15 at 10:55

What is the error you have encountered? Can you post that please – Pogrindis Aug 13 '15 at 12:54

3 Answers

Maybe you should consider Passport or another module. But you can do something like this:

```javascript
app.post('/signup', function (req, res, next) {
    var user = {
        Name: req.body.name,
        Email: req.body.email,
        Pass: req.body.pass,
        Num: req.body.num
    };
    var UserReg = mongoose.model('UserReg', RegSchema);
    UserReg.create(user, function(err, newUser) {
        if(err) return next(err);
        req.session.user = email;
        return res.send('Logged In!');
    });
});

app.post('/login', function (req, res, next) {
    var email = req.body.email;
    var pass = req.body.pass;

    User.findOne({Email: email, Pass: pass}, function(err, user) {
        if(err) return next(err);
        if(!user) return res.send('Not logged in!');

        req.session.user = email;
        return res.send('Logged In!');
    });
});

app.get('/logout', function (req, res) {
    req.session.user = null;
});
```

Then you should have a middleware to handle authentication

```javascript
function isLoggedIn (req, res, next) {
    if (!(req.session && req.session.user)) {
        return res.send('Not logged in!');
    }
    next();
}
```

And use it on the private routes

```javascript
app.get("/api", isLoggedIn, function (req, res) {
    //Something private
})
```

edited Aug 13 '15 at 14:40   answered Aug 13 '15 at 14:35

Rafael P. Miranda
435   3   14

Here is a nice tutorial how to make what you want using very useful module **passport**. Also you will have a quick look at **Jade** template engine which can be useful in your further learning of creating **express** apps.

answered Aug 13 '15 at 11:59

Alexander Korzhykov
170   7

2   but I don't want to use Jade and passport :/ I want to make it "Native" –   Vatex Official   Aug 13 '15 at 12:14

also looking for workable auth example, the sample under the link is not working... just checked out from git , set local mongo and run - no any activity on sighup and login buttons.... – amigo421 Dec 27 '16 at 12:03

check this tutorial...you can ignore Angular and mongojs if you want:
http://www.phloxblog.in/single-page-application-angular-js-node-js-mongodb-mongojs-module/#.Vc20OXW1Gkq

answered Aug 14 '15 at 9:27

Subham
316   2   13