

## ARTIGO: 11560

### Node + Express + Passport

Configuração básica do Node usando Express com Passport para autenticação

#### app.coffee

```
crypto = require('crypto')
passport = require('passport')
LocalStrategy = require('passport-local')
flash = require('connect-flash')

login = require('./routes/login')
logout = require('./routes/logout')

app.use(flash())
app.use(passport.initialize())
app.use(passport.session())

app.use('/login', login)
app.use('/logout', logout)

passport.serializeUser (user, done) ->
  done(null, user._id)

passport.deserializeUser (id, done) ->
  db.Usuario.findById id, (err, user) ->
    done(null, user)

passport.use new LocalStrategy {passReqToCallback : true}, (req, username, password, done) ->

  if(!username || !password)
    return done(err)

  db.Usuario.findOne {'username': username}, (err, user) ->
    if err
      return done(err)

    if !user
      return done(null, false, req.flash('error', 'Usuário ou senha inválidos'))

    password = crypto.createHash('sha1').update(password).digest('hex')

    if password == user.password
      return done(null, user)
    else
      return done(null, false)
```

#### loggedin.coffee

```
isLoggedIn = (req, res, next) ->
  if req.isAuthenticated()
    return next()
  res.redirect('/login')

module.exports.isLoggedIn = isLoggedIn
```

#### login.coffee

```
express = require('express')
passport = require('passport')
router = express.Router()

router.get '/', (req, res) ->
  res.render('login/index', {message: req.flash('error')})

router.post '/', passport.authenticate('local', { successRedirect: '/home', failureRedirect: '/login', failureFlash : true})
```

```
module.exports = router
```

## logout.coffee

```
express = require('express')
passport = require('passport')
router = express.Router()

router.get '/', (req, res) ->
  req.logout()
  res.redirect('/login')

module.exports = router
```

## Usando a autenticação

```
router.get '/:id', isLoggedIn, (req, res) ->

  // req.user
```