TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN

PHÁT TRIỂN ỨNG DỤNG WEB VỚI NODEJS (502070)

LAB8/9 – NODEJS vs MongoDB

TDTU

Express database integration (expressjs.com)

- Cassandra
- Couchbase
- CouchDB
- LevelDB
- MySQL
- **MongoDB**
- Neo4j
- Oracle
- PostgreSQL
- Redis
- SQL Server
- SQLite
- Elasticsearch

Express database integration (expressjs.com)

```
$ npm install mongodb
```

Example (v2.*)

```
const MongoClient = require('mongodb').MongoClient

MongoClient.connect('mongodb://localhost:27017/animals', (err, db) => {
  if (err) throw err

  db.collection('mammals').find().toArray((err, result) => {
    if (err) throw err

    console.log(result)
  })
})
```

Express database integration (expressjs.com)

```
$ npm install mongodb
```

Example (v3.*)

```
const MongoClient = require('mongodb').MongoClient

MongoClient.connect('mongodb://localhost:27017/animals', (err, client) => {
  if (err) throw err

  const db = client.db('animals')

  db.collection('mammals').find().toArray((err, result) => {
    if (err) throw err

    console.log(result)
  })
})
```

[Introduction to MongoDB — MongoDB Manual]

MongoDB is an open source NoSQL database management program. NoSQL is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. MongoDB is a tool that can manage document-oriented information, store or retrieve information.

mongoose

https://mongoosejs.com

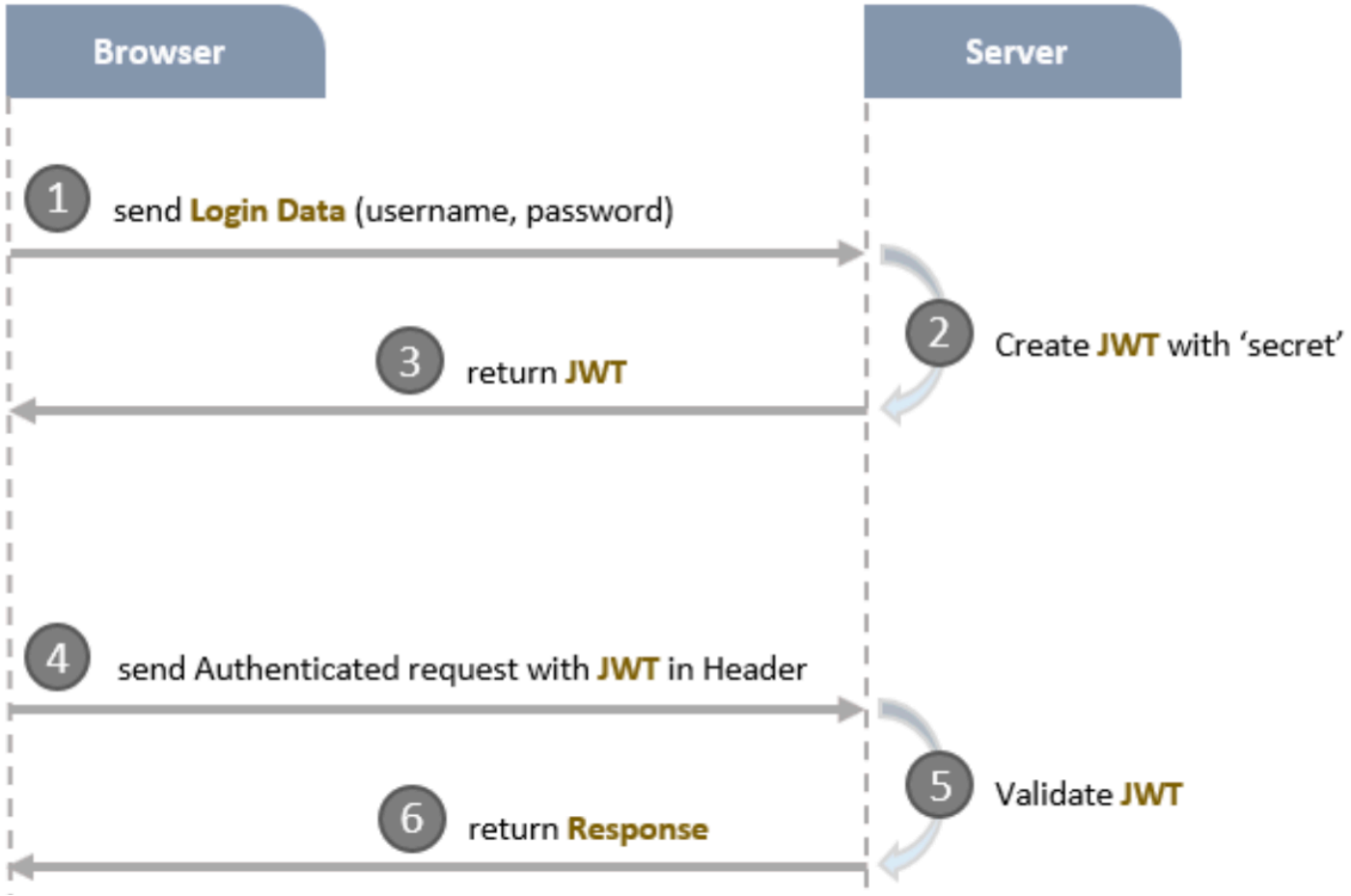
elegant mongodb object modeling for node.js

```
$ npm install mongoose --save
```

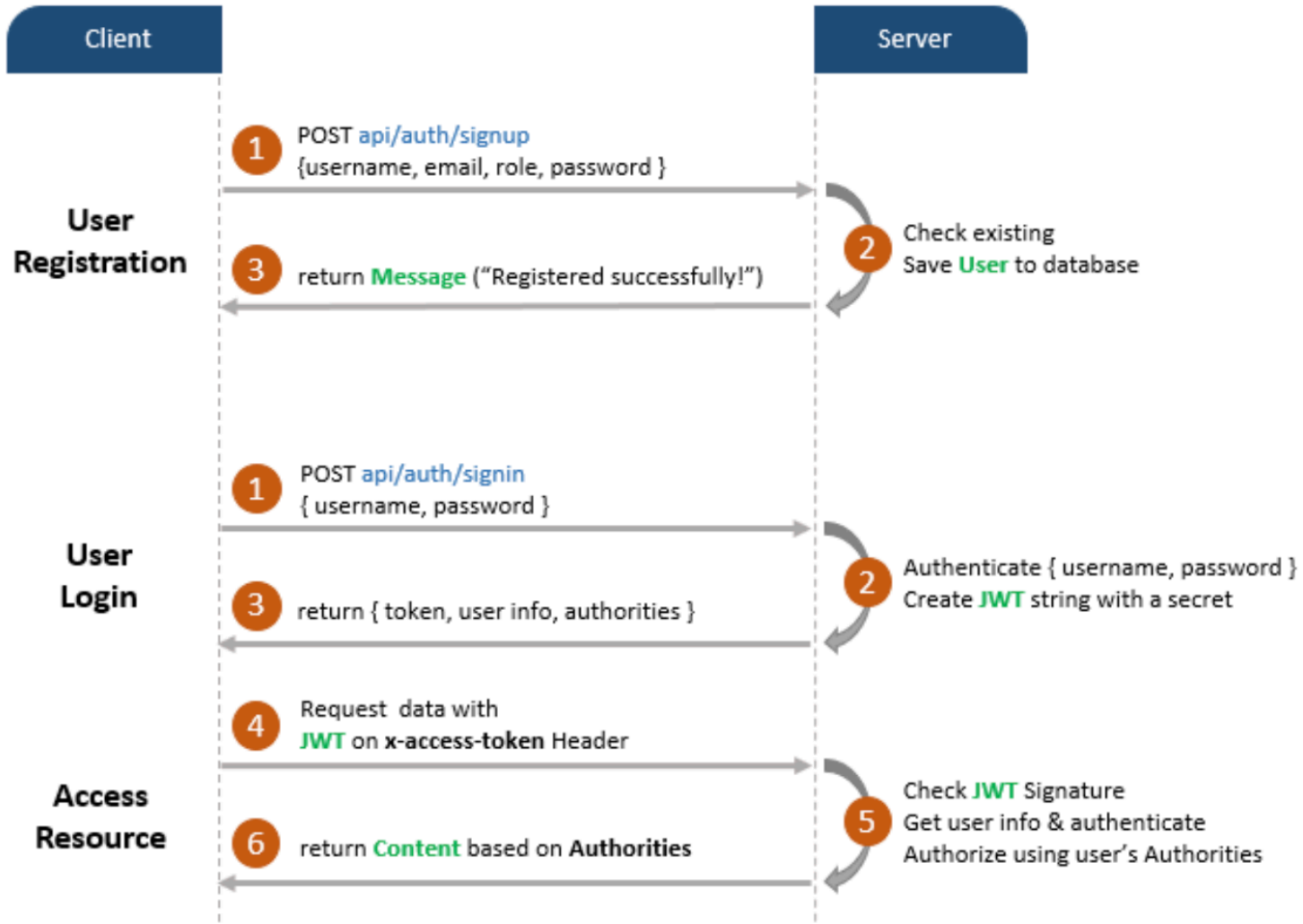
```
const mongoose = require('mongoose');

main().catch(err => console.log(err));

async function main() {
  await mongoose.connect('mongodb://localhost:27017/test');
}
```

```
const mongoose = require('mongoose');
mongoose.connect('mongodb://localhost:27017/test');

const Cat = mongoose.model('Cat', { name: String });

const kitty = new Cat({ name: 'Zildjian' });
kitty.save().then(() => console.log('meow'));
```

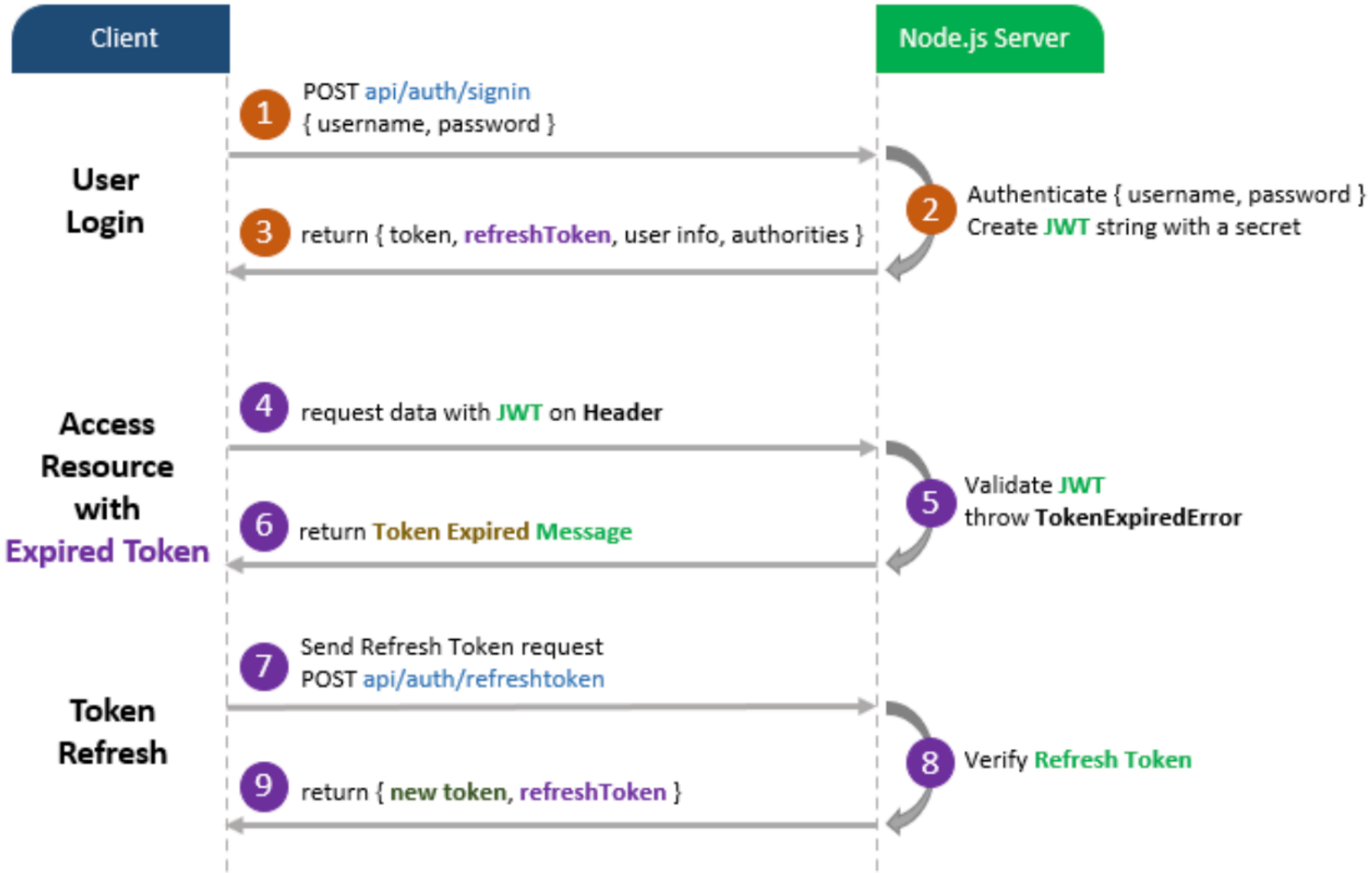
npm install jsonwebtoken
express-jwt - npm (npmjs.com)



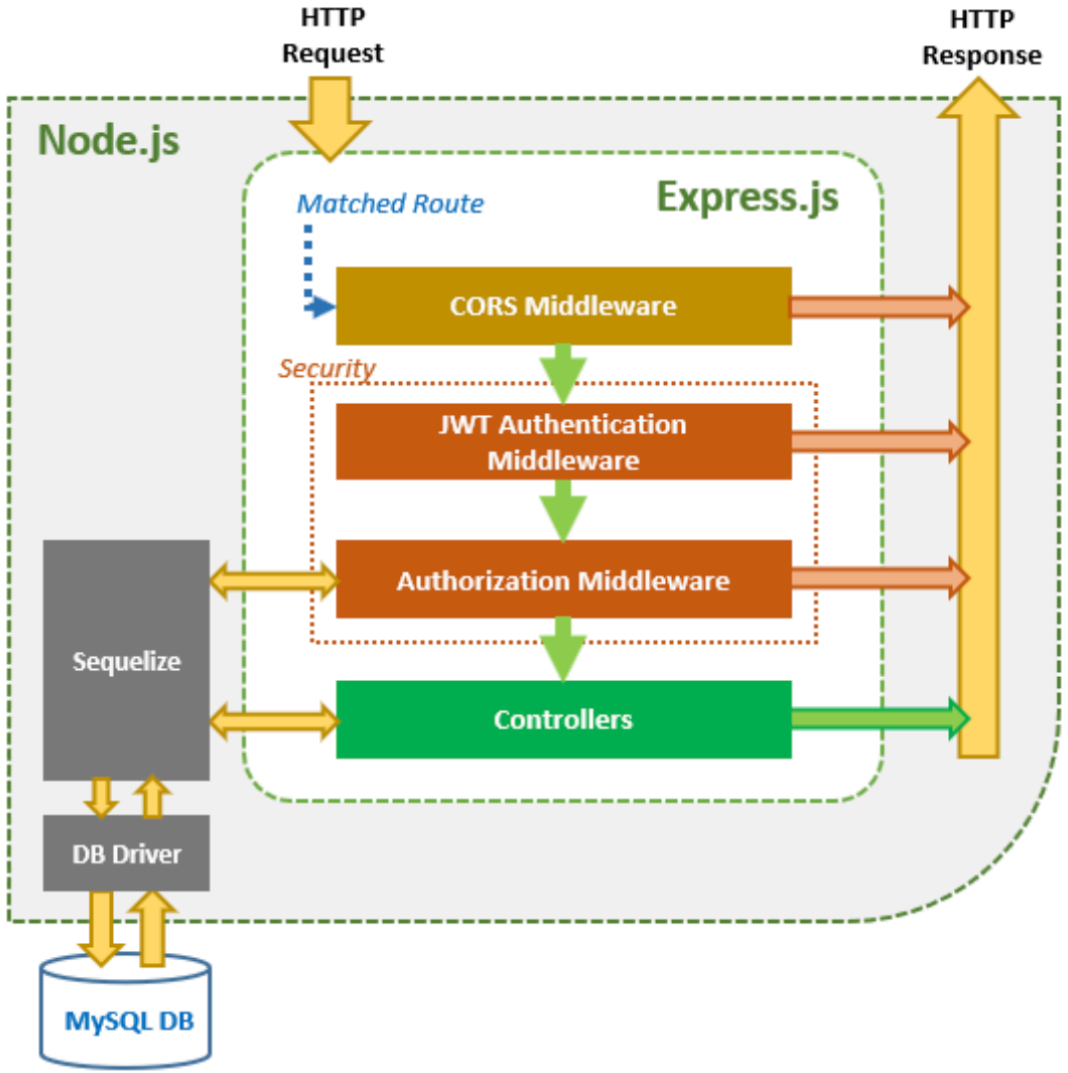
npm install jsonwebtoken
express-jwt - npm (npmjs.com)



npm install jsonwebtoken
express-jwt - npm (npmjs.com)



npm install jsonwebtoken
express-jwt - npm (npmjs.com)

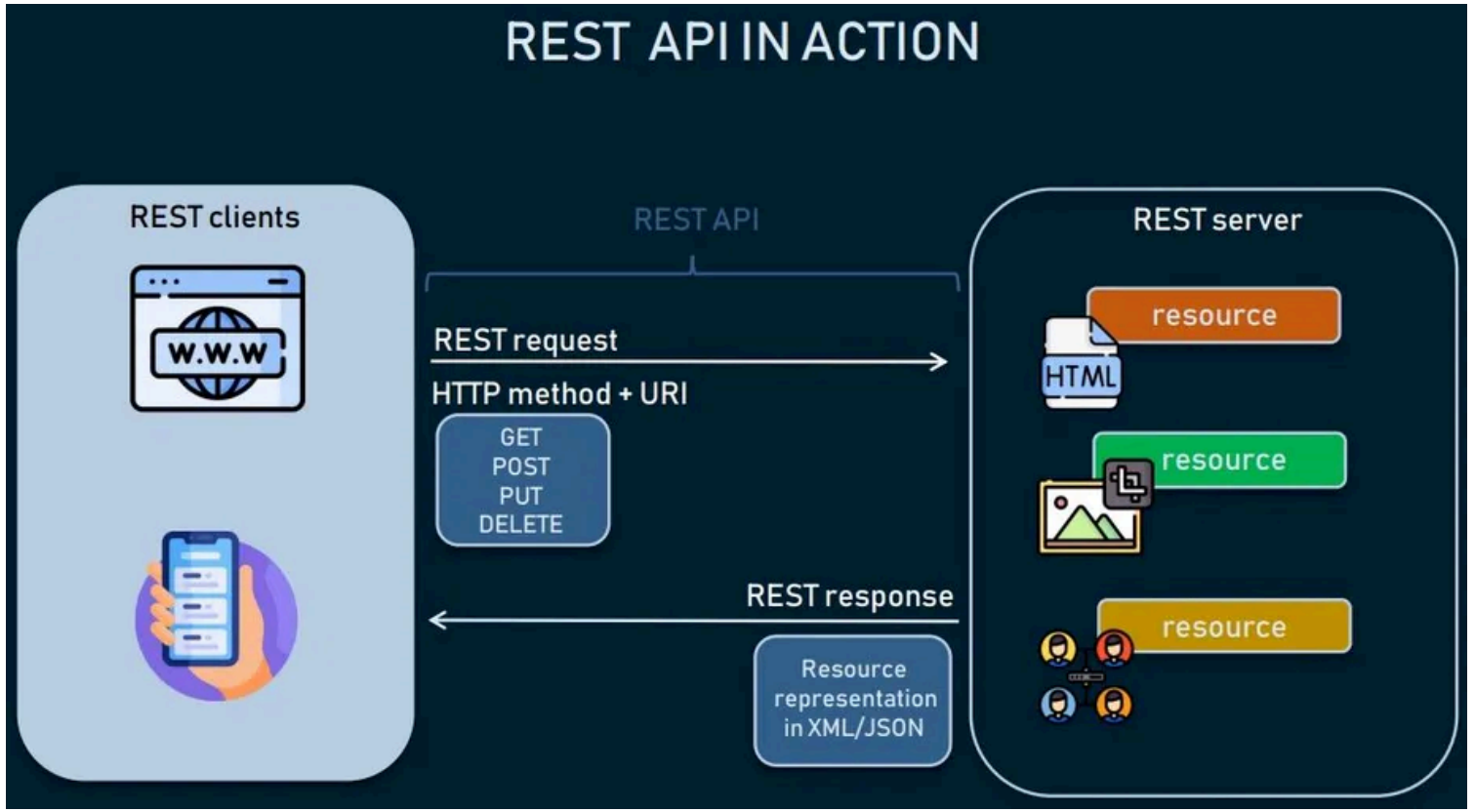


Project structure (recommended)

- **config**
  - configure database…
  - configure Auth Key
- **routes**
  - *auth.routes.js*: POST signup & signin
  - *user.routes.js*: GET public & protected resources
- **middlewares**
  - *verifySignUp.js*: check signup information
  - *authJwt.js*: verify Token, check User roles in database
- **controllers**
  - *auth.controller.js*: handle signup & signin actions
  - *user.controller.js*: return public & protected content
- **models**
  - *user.model.js*
  - *role.model.js*

server.js: import and initialize necessary modules and routes, listen for connections.

A REST API (also known as **RESTful API**) is an application programming interface (API or web API) that conforms to the constraints of REST architectural style and allows for interaction with RESTful web services. REST stands for **representational state transfer** and was created by computer scientist Roy Fielding.



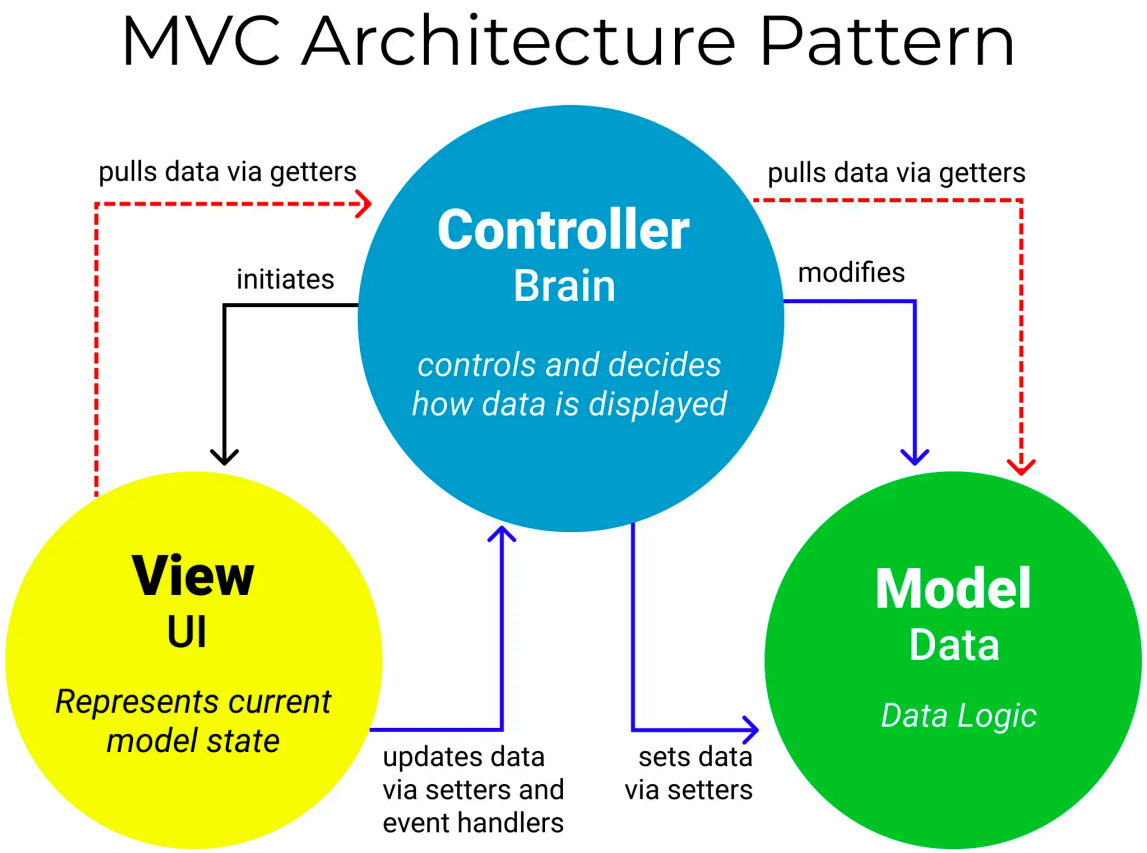
Express cors middleware (expressjs.com)

```
var express = require('express')
var cors = require('cors')
var app = express()

var corsOptions = {
  origin: 'http://example.com',
  optionsSuccessStatus: 200 // some legacy browsers (IE11, various SmartTVs)
choke on 204
}

app.get('/products/:id', cors(corsOptions), function (req, res, next) {
  res.json({msg: 'This is CORS-enabled for only example.com.'})
})

app.listen(80, function () {
  console.log('CORS-enabled web server listening on port 80')
})
```

Express routing (expressjs.com)

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

// middleware that is specific to this router
router.use((req, res, next) => {
  console.log('Time: ', Date.now())
  next()
})
// define the home page route
router.get('/', (req, res) => {
  res.send('Birds home page')
})
// define the about route
router.get('/about', (req, res) => {
  res.send('About birds')
})

module.exports = router
```

Design Patterns - MVC Pattern (tutorialspoint.com)



Thank you