**1. In your own words, describe what a middleware is and why it is important.**

Yes it goes without saying that middleware are very important. WHY? They are important because we can run set of statement before the actual handler is executed . For example we can use Middlewares to write logs and Authentication.

**2. Please describe the architecture of your middleware solution and why it is implemented in such a way**

project/

controllers/

helpers/

**middlewares/**

**auth.js**

models/  
 public/

I will organize **middlewares** in a separate folder at project level because there can be multiple middlewares and different middleware for different controllers.

**3. How would you handle cross-origin resource sharing in your solution?**

I would maintain a list of origins that are allowed and use the following code to implement the CORS in my project:

**app.use(function (req, res, next) {**

**var allowedOrigins = ['http://127.0.0.1:8020', 'http://localhost:8020', 'http://127.0.0.1:9000', 'http://localhost:9000'];**

**var origin = req.headers.origin;**

**if (allowedOrigins.indexOf(origin) > -1) {**

**res.setHeader('Access-Control-Allow-Origin', origin);**

**}**

**res.header('Access-Control-Allow-Methods', 'GET, OPTIONS');**

**res.header('Access-Control-Allow-Headers', 'Content-Type, Authorization');**

**res.header('Access-Control-Allow-Credentials', true);**

**return next();**

**});**

**4. Based on your opinion, what is the best approach in setting up the middleware environments?**

In my opinion middleware should be stand alone and configurable. Middleware should be managed separately and work like plug and play.

**5. How does one test a middleware to make sure it is running according to specifications?**

One can use node js unit test frameworks (e.g. MOCHA) and mock the behavior of the end user to validate if the middleware are working according to the specifications.

**6. What is a secure middleware and how do you ensure you have one?**

Secure middleware is one that protect application from attacks like **cross site scripting, mime sniffing** etc..

***HELMET*** is one of the most popular secure middleware that is used widely

**7. What are some approaches to improving middleware performance?**

Apply middlewares on specific route and handlers instead of the root(/). Do not overfill a middleware with a lot of responsibilities instead of this divide the work into multiple middlewares and it is quite possible that not every handler needs all the statements present into a single middleware so make middleware controller specific.