

BACKEND WS-8 – SOLUTIONS

29.11.2023

1- PRE MIDDLEWARE NEDIR VE NASIL CALISIR?

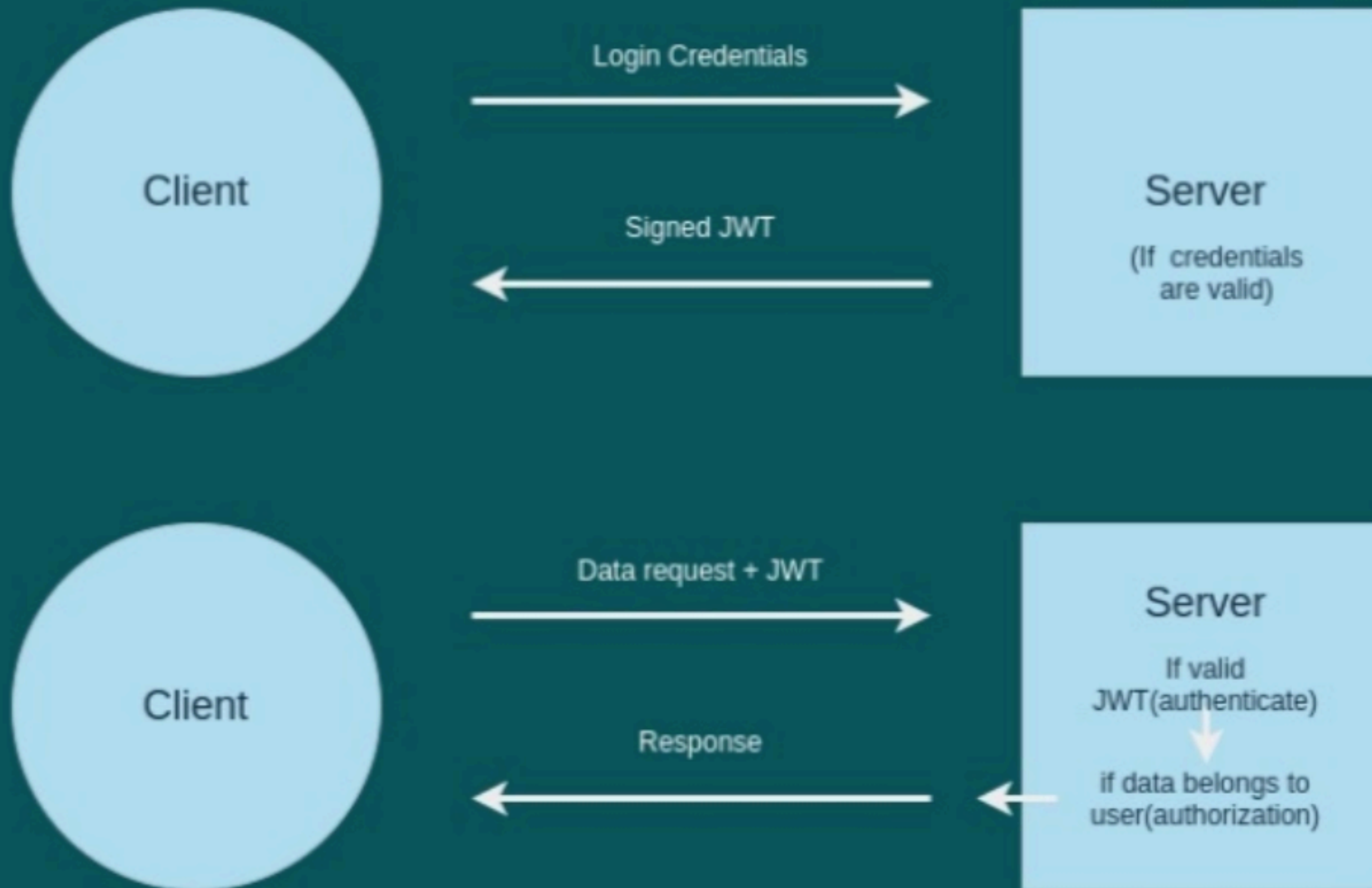
- <https://techinsights.manisuec.com/mongodb/mongoose-pre-and-post-hooks-middlewares/>
- <https://medium.com/swlh/need-of-mongoose-middleware-in-authentication-3ddc70a993f>
- **Pre** middleware gets executed before the operation happens. For instance, a **pre-save** middleware will get executed before the saving of the document. This functionality makes **pre** middleware perfect for more complex validations and default values assignment.
- A **pre** middleware is defined using the **pre()** method of the schema object, so validating your model using a **pre** middleware will look like the following code snippet:

```
UserSchema.pre('save', function(next) {  
  if (...) {  
    next()  
  } else {  
    next(new Error('An Error Occured'));  
  }  
});
```

2- SIMPLE TOKEN NEDİR, NASIL CALISIR?

- Express.js'deki simple token tabanlı kimlik doğrulama sistemi, oturum açtığında kimliği doğrulanmış bir kullanıcıya bir token verilmesini içerir. Bu token daha sonra kullanıcının kimliğini doğrulamak için sonraki isteklerle birlikte gönderilir. Sunucu belirteci doğrular ve gerçekliğine göre erişime izin verir veya reddeder. API uç noktalarının güvenliğini sağlamanın basit bir yoludur.

3- EXPRESS.JS'DE AUTHENTICATION ICIN TOKENLER NASIL ÜRETİLİR?



4- SIMPLE TOKEN'LA YAPILAN AUTHENTICATION'IN NORMAL USERNAME VE PASSWORD ILE YAPILANA KARŞI AVANTAJLARI NELERDİR?

- Simple token tabanlı kimlik doğrulama, geleneksel yöntemlere göre daha iyi güvenlik, ölçeklenebilirlik ve esneklik gibi çeşitli avantajlar sunar. Hassas kullanıcı parolalarının sunucuda saklanması ihtiyacını ortadan kaldırır, veri ihlali riskini azaltır ve oturum yönetimine gerek kalmadan istemcilerin sorunsuz API erişimine olanak tanır.

5-

- Token tabanlı kimlik doğrulama, istemci ile sunucu arasında güvenli iletişim kurmak için kullanılan bir yöntemdir. Kullanıcılar oturum açmak için _____ ve _____ yerine bir token alır ve kimliklerini bu token ile doğrularlar.

5- CEVAP

- Token tabanlı kimlik doğrulama, istemci ile sunucu arasında güvenli iletişim kurmak için kullanılan bir yöntemdir. Kullanıcılar oturum açmak için ____username__ ve ____password__ yerine bir token alır ve kimliklerini bu token ile doğrularlar.

6-

- Express.js'de, belirli kaynaklar için kullanıcı erişim izinlerini kontrol etmek için kullanılan bir ara katman(middleware) türü _____ ara katmanı olarak adlandırılır. Bu ara yazılım isteği inceler, kullanıcının izinlerini kontrol eder ve isteği, islemeden _____ veya _____ değiştirebilir.

6- CEVAP

- Express.js'de, belirli kaynaklar için kullanıcı erişim izinlerini kontrol etmek için kullanılan bir ara katman(middleware) türü permissions ara katmanı olarak adlandırılır. Bu ara yazılım isteği inceler, kullanıcının izinlerini kontrol eder ve isteği, işlemeyen önce veya sonra değiştirebilir.

7-

- Express.js'de bir isteği işlemekten önce veya sonra ön işleme tabi tutmak için kullanılan bir ara yazılım (middleware) türü _____ ara yazılımı olarak bilinir. Örneğin, bu ara yazılım gelen verileri doğrulamak, kimlik doğrulaması yapmak veya isteği bir şekilde kısıtlamak için kullanılabilir.

7-CEVAP

- Express.js'de bir isteği işlemekten önce veya sonra ön işleme tabi tutmak için kullanılan bir ara yazılım (middleware) türü `__pre-middleware__` ara yazılımı olarak bilinir. Örneğin, bu ara yazılım gelen verileri doğrulamak, kimlik doğrulaması yapmak veya isteği bir şekilde kısıtlamak için kullanılabilir.

8- EXPRESS.JS KULLANARAK E-POSTA GÖNDERMEK İÇİN HANGİ YÖNTEMLERİ VE KÜTÜPHANELERİ KULLANABİLİRSİNİZ?

► Node.js ile e-posta göndermek için 3 seçenek:

1. Simple Mail Transfer Protocol (SMTP)(Nodemailer)

Basit Posta Aktarım Protokolü (SMTP), ağlar üzerinden giden e-postalar göndermek için kullanılan bir teknolojidir ve en yaygın aktarım yöntemidir. Bir sunucudan diğerine e-posta göndermek için bir aktarım hizmeti görevi görür. Gmail gibi bir e-posta istemcisi kullanarak bir arkadaşınıza e-posta gönderdiğinizde, giden (SMTP) bir sunucu bunu alır ve arkadaşınızın alıcı sunucusuna bağlanır. İki sunucu SMTP protokolü tarafından tanımlanan yönergeleri kullanarak iletişim kurar, alıcının kim olduğunu ve gelen postayı nasıl alabileceğini belirler. E-posta istemcileri genellikle e-posta dağıtımına yardımcı olmak için kendileriyle ilişkili bir SMTP sunucusuna sahiptir.

2. Bir Email API'si kullanmak

İşlemsel e-posta API'leri, hızlı bir şekilde entegre edilebilen, yüksek hacimli gönderimi destekleyebilen ve zengin işlevsellik sunan güvenilir bir hizmete ihtiyaç duyduğunuzda kullanışlıdır. Piyasada birçok e-posta hizmeti bulunmaktadır. En popüler olanları arasında Amazon SES, Postmark, SparkPost, SendGrid, Mailgun ve Mailchimp Transactional (eski adıyla Mandrill) bulunmaktadır. Bunların hepsi ücretli hizmetlerdir, ancak çoğu ücretsiz veya düşük maliyetli tanıtım planları sunar.

3- Bir çok kanallı bildirim servisi kullanmak

Courier gibi bir bildirim hizmeti, tipik bildirimlerde yer alan tüm mantığı desteklemek için güçlü API ilkeleri sunar ve tek bir API kullanarak bir dizi farklı kanaldaki kullanıcılara ulaşmanıza olanak tanır. Genellikle her kanal için kendi sağlayıcınızı getirmenize izin verirler; e-posta söz konusu olduğunda, bu kendi SMTP sunucunuz veya barındırılan bir e-posta API'si olabilir. Bir bildirim hizmeti aynı kanal için farklı sağlayıcılara yük devretmenize de izin verebilir

Bir bildirim hizmetiyle, kodunuza dokunmanıza gerek kalmadan kolayca daha fazla kanal ekleyebilir ve hatta e-posta hizmet sağlayıcınızı değiştirebilirsiniz. Kullanıcıları e-posta, SMS, push veya Slack ve WhatsApp gibi sohbet uygulamaları üzerinden bilgilendirmek istiyorsanız, bunu tek bir hamlede yapabilirsiniz.

Özellikle Courier, işlemsel bir e-posta hizmeti ile elde edeceğinizin ötesinde size ek işlevsellik sunar. E-postalarınızı esnek bir görsel ve kod düzenleyicide tasarlayabilir, teslimat kuralları belirleyebilir ve basit iş akışları oluşturabilir ve teslimat durumunu gerçek zamanlı olarak izleyebilirsiniz.

<https://www.courier.com/blog/how-to-send-emails-with-node-js/>

9- BİR EXPRESS.JS UYGULAMASINDA DOSYA YÜKLEME İŞLEMİNİ NASIL GERÇEKLEŞTİREBİLİRSİNİZ? BU AMAÇ İÇİN YAYGIN OLARAK HANGİ ARA KATMAN YAZILIMLARI VEYA KÜTÜPHANELER KULLANILIR?

- Multer, connect, body-parser vb. gibi kullanılabilecek birçok ara yazılım vardır. En çok kullanılan yol multer.
- <https://www.freecodecamp.org/news/simplify-your-file-upload-process-in-express-js/#:~:text=For%20file%20uploads,%20you%20should,multipart/form-data>".