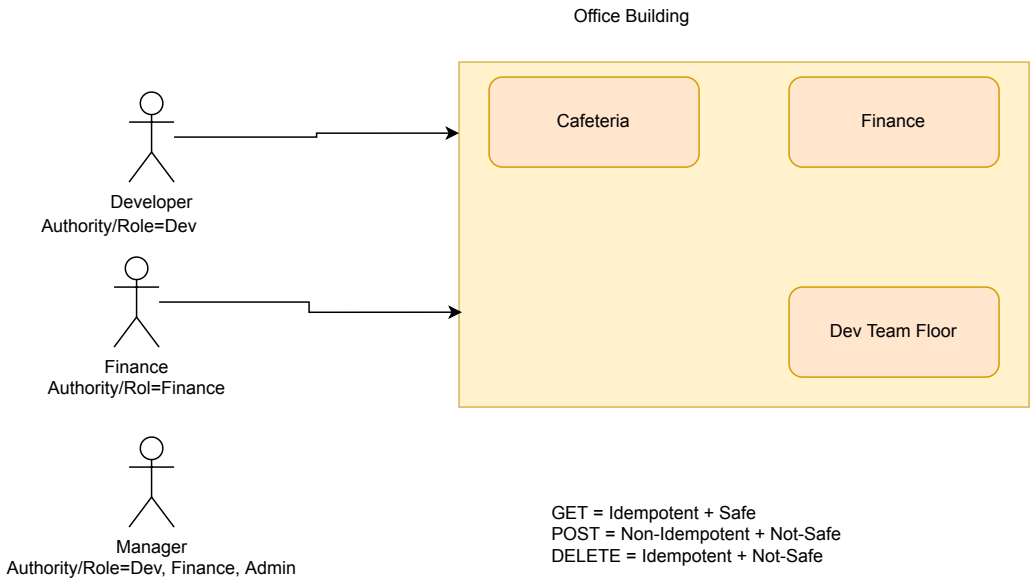
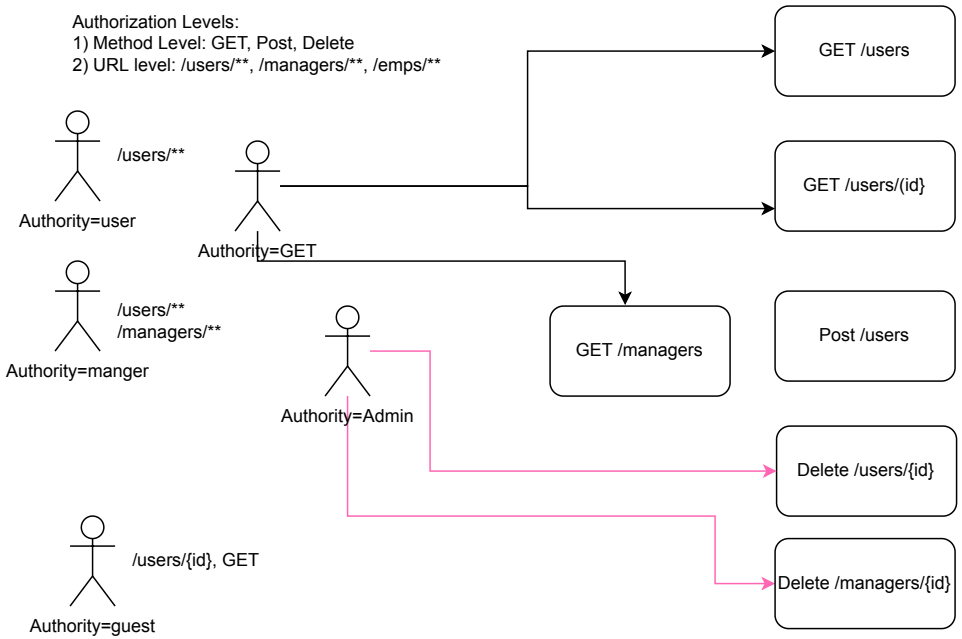
Security

Authentication vs Authorization

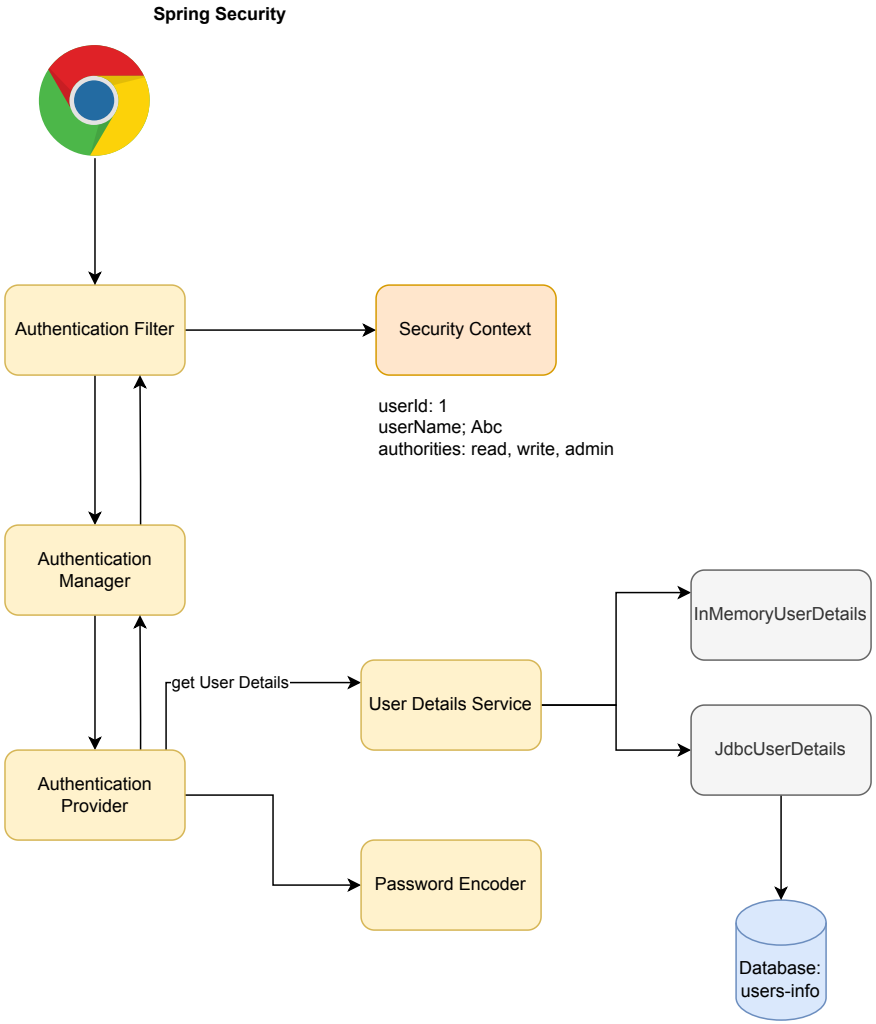


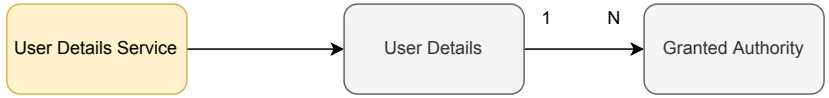
GET = Idempotent + Safe
POST = Non-Idempotent + Not-Safe
DELETE = Idempotent + Not-Safe



Idempotent:
eg. update user_data set name='abc' where name='Abhilash'

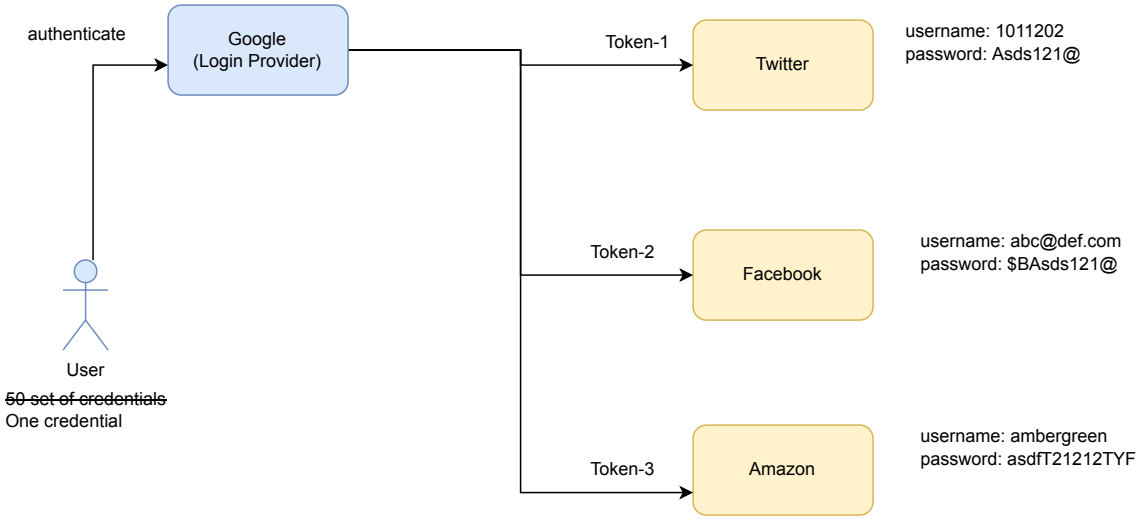
Abhilash -> abc
Amol
Shivam


Spring Security
Authentication Filter
Security Context
userId: 1
userName; Abc
authorities: read, write, admin
Authentication Manager
get User Details
User Details Service
InMemoryUserDetails
JdbcUserDetails
Authentication Provider
Password Encoder
Database: users-info


User Details Service
User Details
1
N
Granted Authority

Single Sign On

(Third Party Login)



Single Sign On Explained

(OAuth2)

OAuth2 Players:

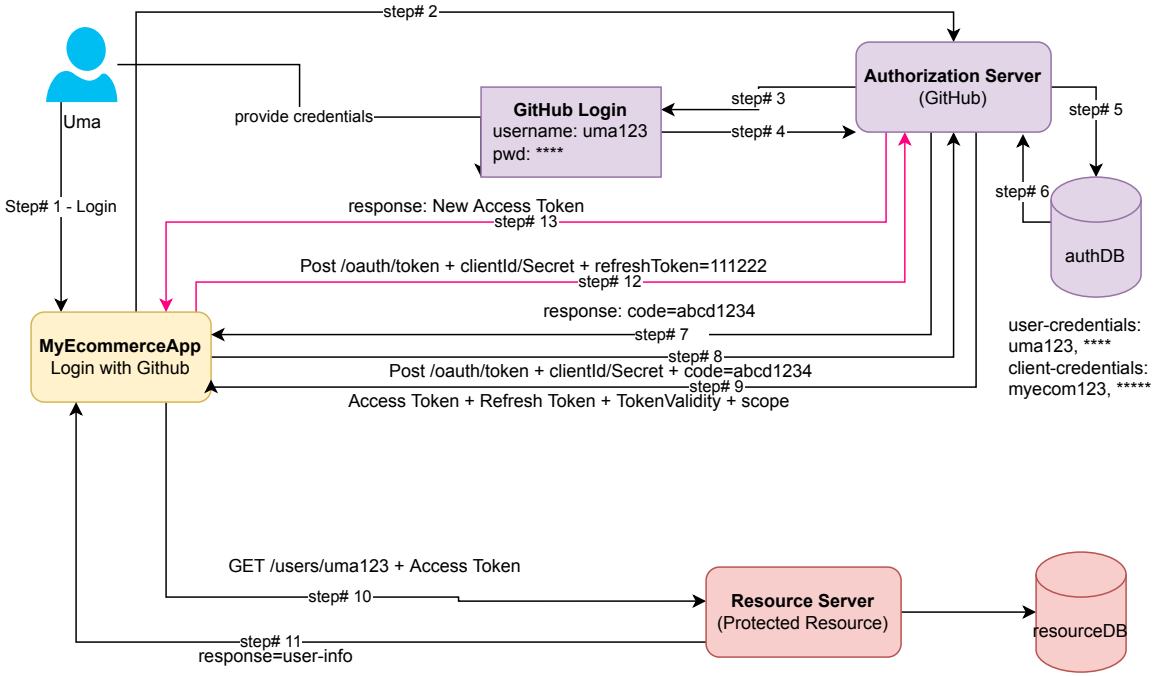
1. Resource Owner (Uma)
2. Client (MyecommerceApp)
3. Authorization Server (GitHub)
4. Resource Server (GitHub/Client)

Authentication:

Human User: Username + Password
Machine User: ClientId + Secret

Grant Types:

1. Authorization Code (sso)
2. Implicit (Deprecated)
3. Password
4. Client Credentials

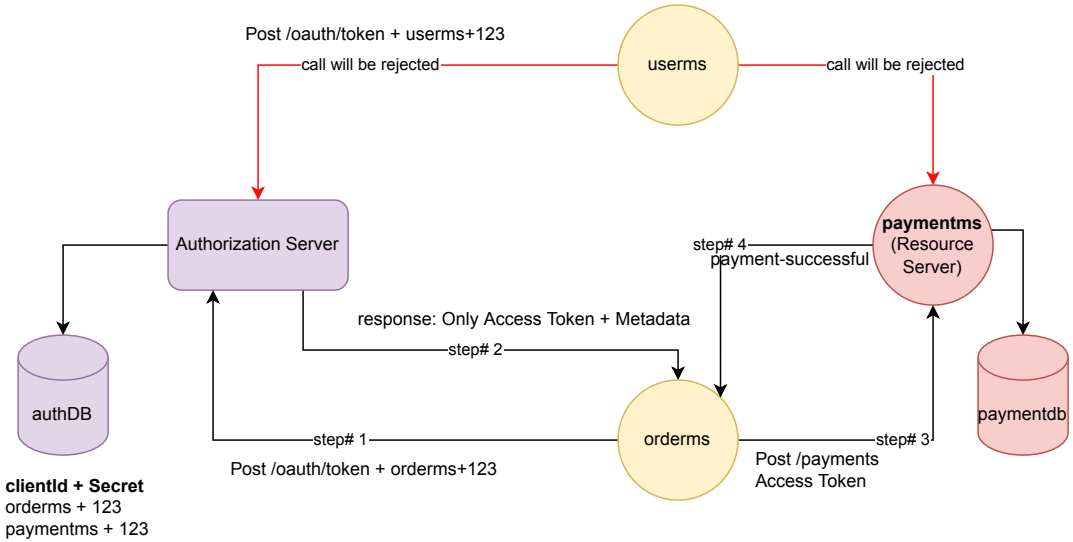


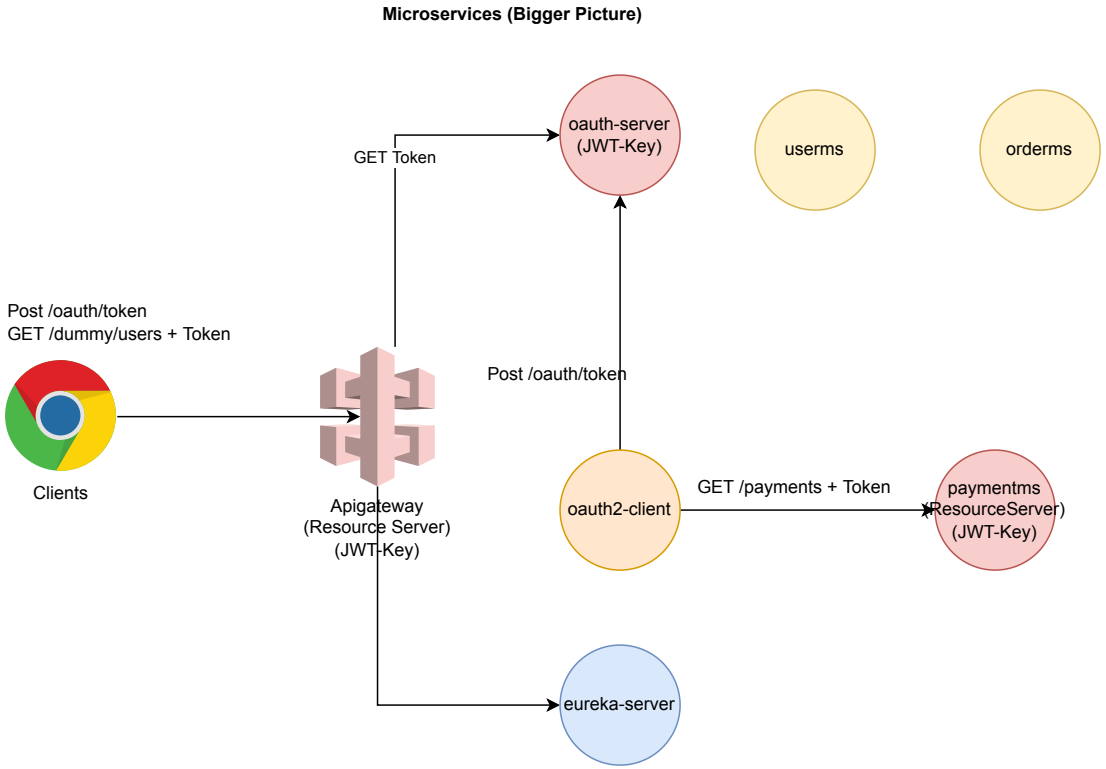
Opaque Token: Step# 9

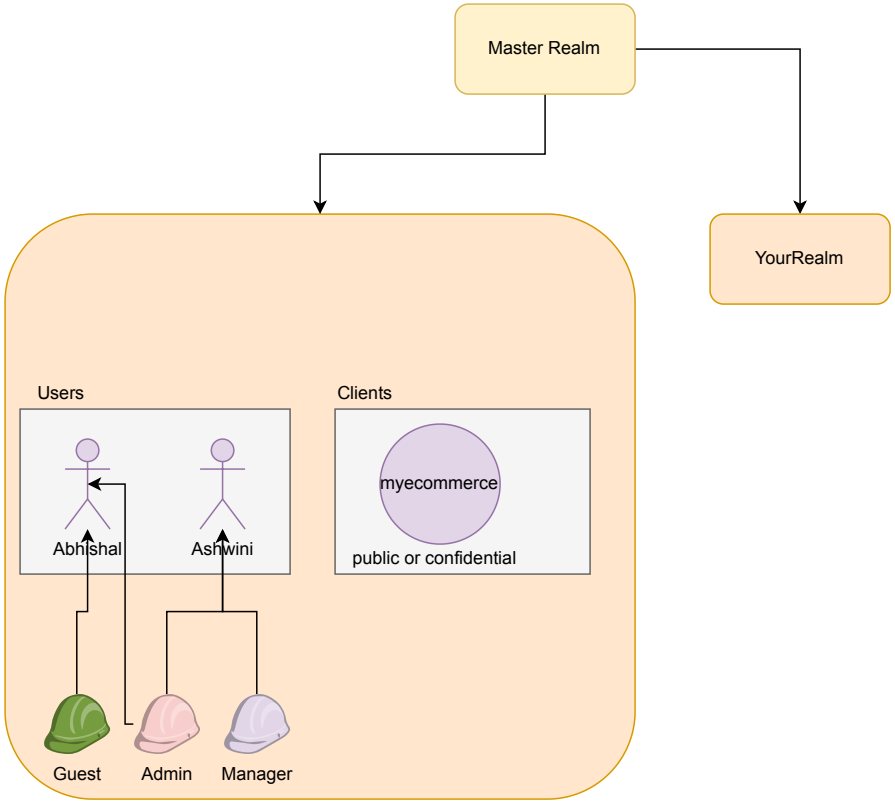
{
"access_token":"3999755e-e987-4a8b-8666-d109ce0261f3",
"refresh_token":"1111111111112222222233333444444443"
"token_type":"bearer",
"expires_in":38,
"scope":"read write"
}

Client Credentials

(Refresh Token not supported)




Microservices (Bigger Picture)
oauth-server
(JWT-Key)
userms
orderms
GET Token
Post /oauth/token
GET /dummy/users + Token
Post /oauth/token
Clients
Apigateway
(Resource Server)
(JWT-Key)
oauth2-client
GET /payments + Token
paymentms
ResourceServer)
(JWT-Key)
eureka-server


Master Realm
YourRealm
Users
Clients
myecommerce
public or confidential
Abhishal
Ashwini
Guest
Admin
Manager