http://localhost:8400/auth/swagger-ui.html

http://localhost:8200/pensioner/swagger-ui.html

http://localhost:8300/disbursement/swagger-ui.html

http://localhost:8100/process/swagger-ui.html

Lb – 80

Sec grp – aud 7007

Service -all traffic, all tcp

pension-cluster-890387

AWS\_DEFAULT\_REGION us-east-1 PLAINTEXT REPOSITORY\_URI 156962844638.dkr.ecr.us-east-1.amazonaws.com/audit-management-authorization PLAINTEXT CONTAINER\_NAME aws-auditmanagement-authorization-microservice

890387-pms-auth-task-definition

890387-pms-auth-container

890387-pms-auth-repository

890387-pms-auth-tg

890387-pms-auth-lb

890387-pms-auth-service

pms-disbursement-td-890387

pms-disbursement-container-890387

pms-disbursement-repository-890387

pms-disbursement-tg-890387

pms-disbursement-lb-890387

pms-disbursement-service-890387

pms-pensioner-details-td-890387

pms-pensioner-details-container-890387

pms-pensioner-details-repository-890387

pms-pensioner-details-tg-890387

pms-pensioner-details-lb-890387

pms-pensioner-details-service-890387

pms-portal-td-890387

pms-portal-container-890387

pms-portal-repository-890387

pms-portal-tg-890387

pms-portal-lb-890387

pms-portal-service-890387

pms-process-pension-td-890387

pms-process-pension-container-890387

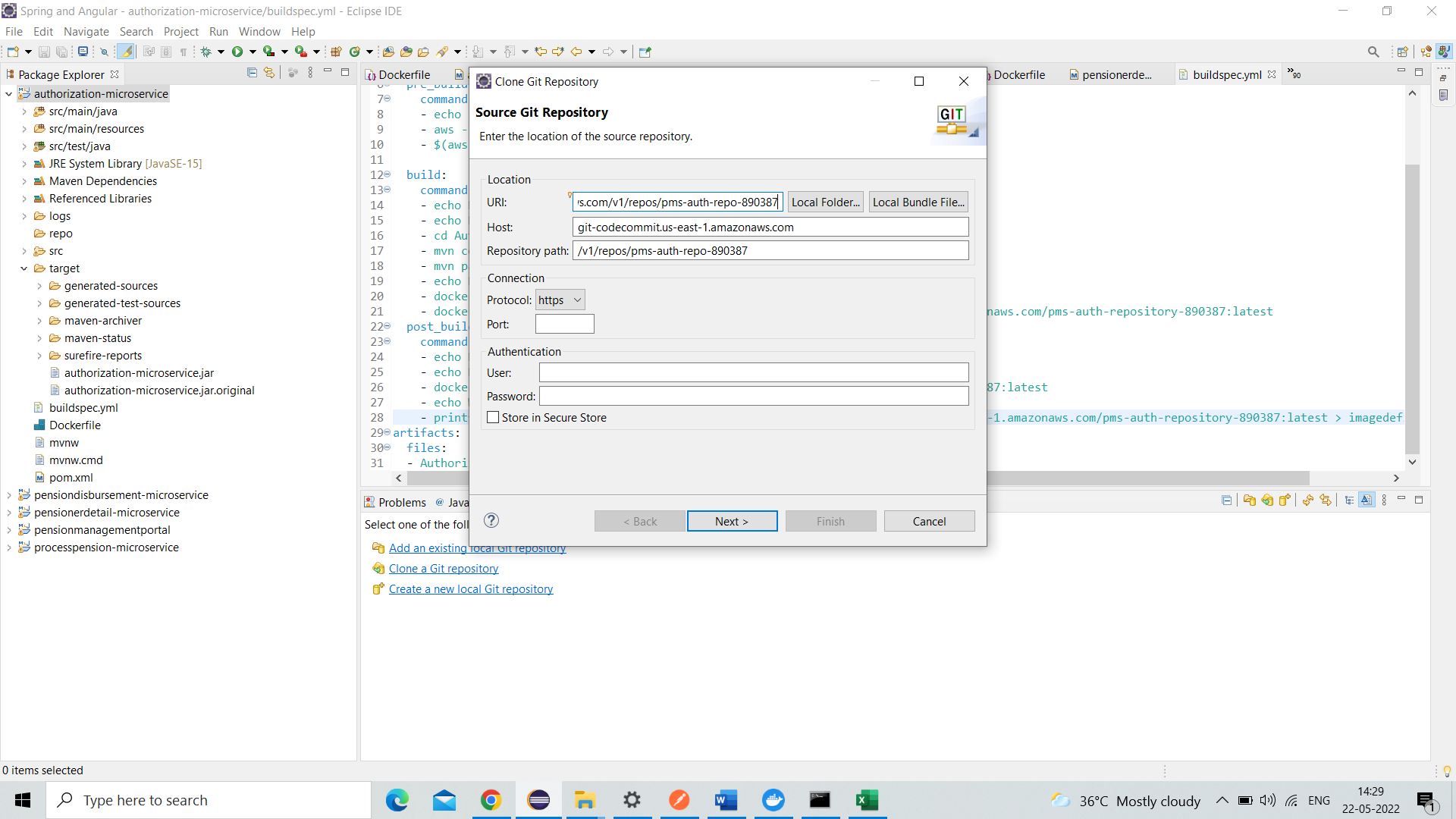
pms-process-pension-repository-890387

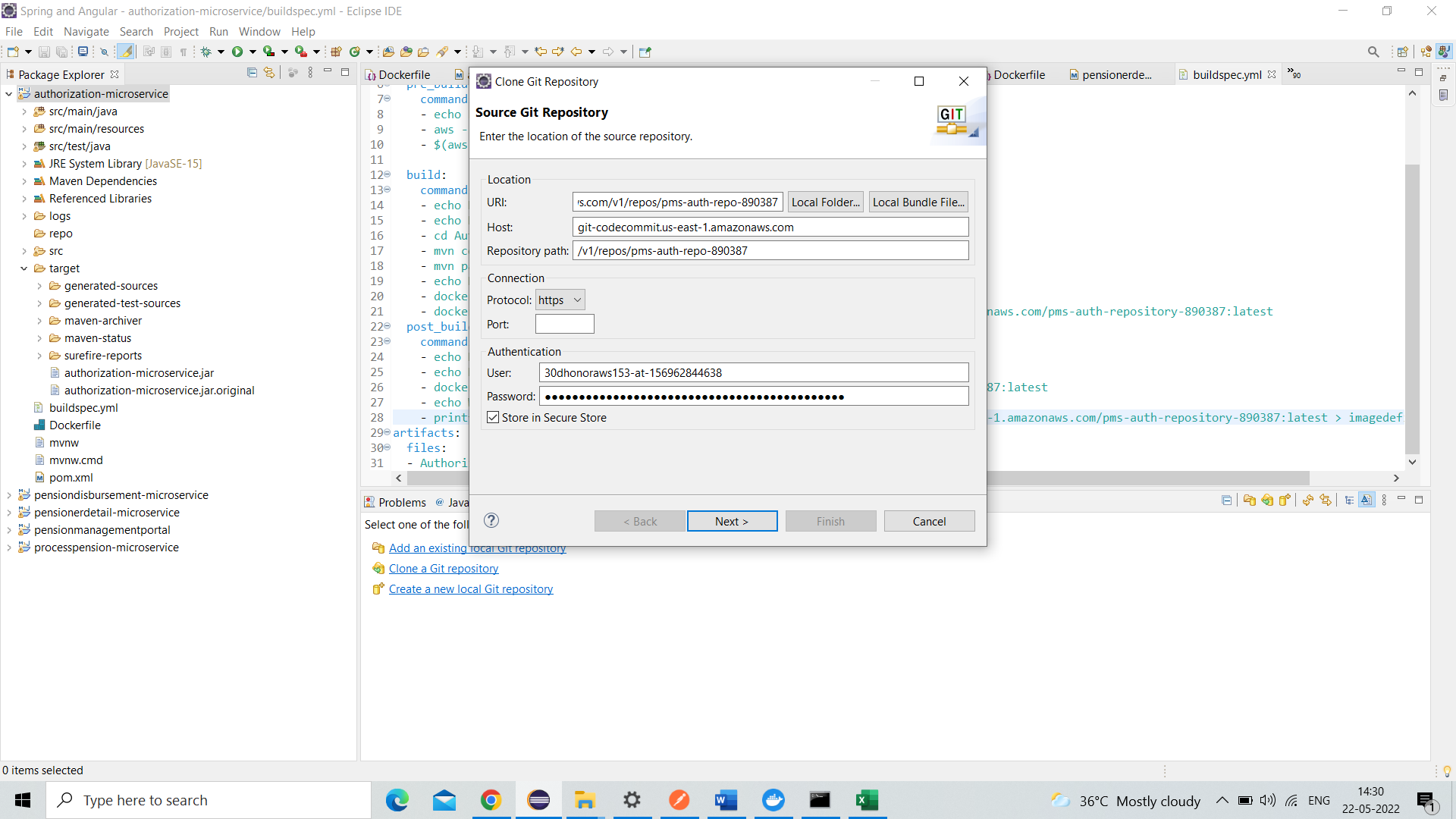
pms-process-pension-tg-890387

pms-process-pension-lb-890387

pms-process-pension-service-890387

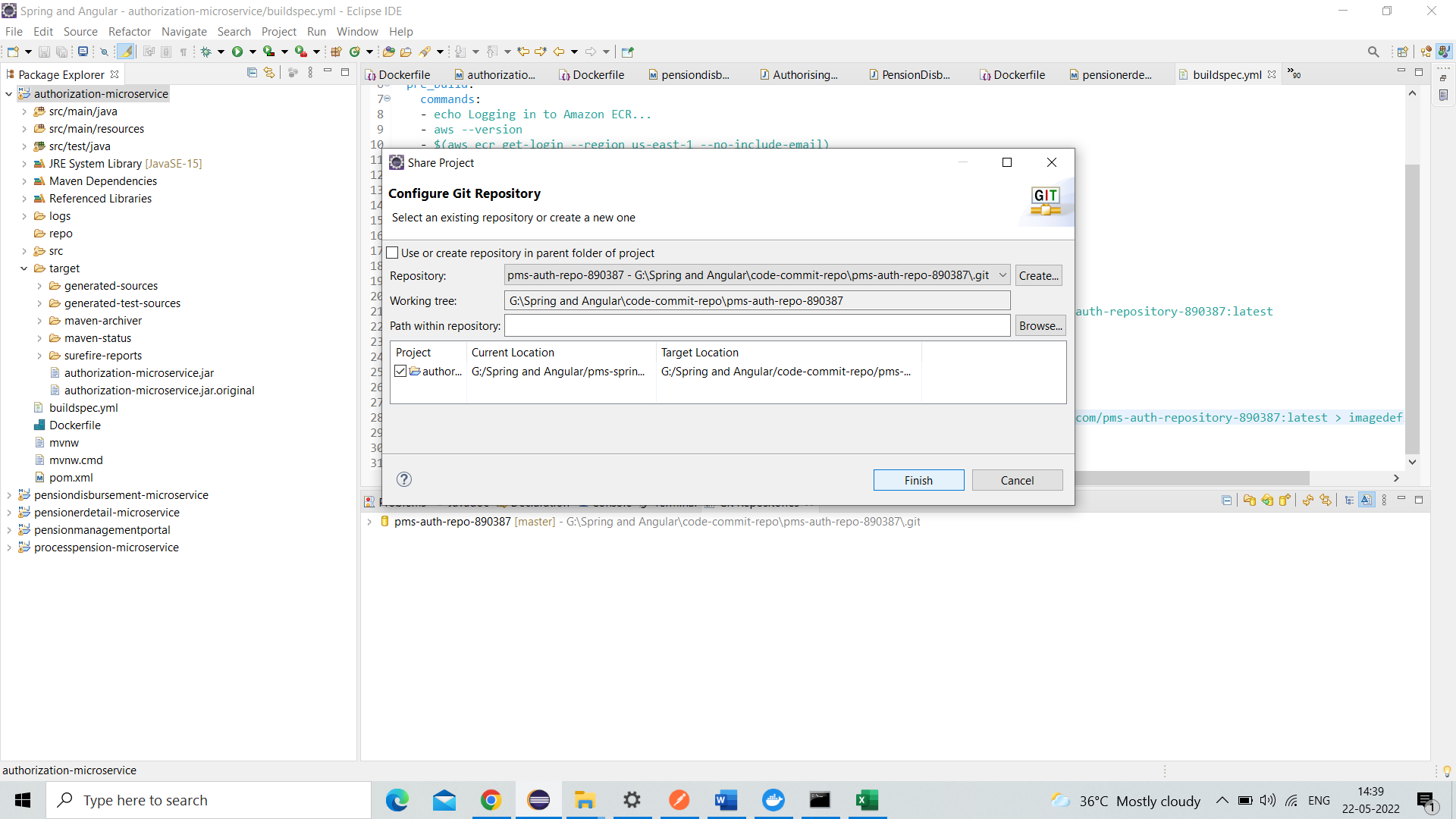
Code commit – create repository – Generate Https git credentials for AWS code commit

Show view – git repositories (in eclipse) – clone a git repository (eclipse) – copy https url from aws code commit repo (<https://git-codecommit.us-east-1.amazonaws.com/v1/repos/pms-auth-repo-890387>) - 



User credentials – from code commit csv file – next (2 times) – select local dir – click finish

Right click proj – team – share proj –



Click finish

ECS - Cluster – without vpc

1. Task definition –fargate - name – without role – task mem 0.5 – task cpu 0.25 –
2. add container – duplicate tab (create repo ) – copy URI (repo) – paste it in container repo- soft limit 128 – app port num – env (cpu units 128 ) – add
3. create – task def compltd – cloud watch cmpltd
4. EC2 – target grp – name – type (ip) – protocol (http) – port -vpc default -create
5. EC2 – load balancer -create – name – internet facing -ipv4 – load bal protocol http – port – select vpc and subnets
6. Create new sg -type http – protocol tcp -port – source custom 0.0.0.0/0
7. Configure routing – existing target grp -create – lb successfully created
8. Cluster – service – fargate -service name – num of task 1 – next step
9. Cluster vpc – select the one which we selected during load balancer -and select multiple subnets-create sg – custom tcp – port – anywhere – save
10. Enabled – select app lb -add lb – create new ( port 8400) – set auto scaling default -create service – view service
11. ECR – repo – image – view push commands – follow the instructions

Container repo img - load balancer

Cluster – task definition –- service – pipeline

pms-authorization-repository

pms-auth-cluster

pms-auth-td

pms-auth-container

pms-auth-service

pms-auth-target-group

pms-auth-load-bal

pms-auth-security

