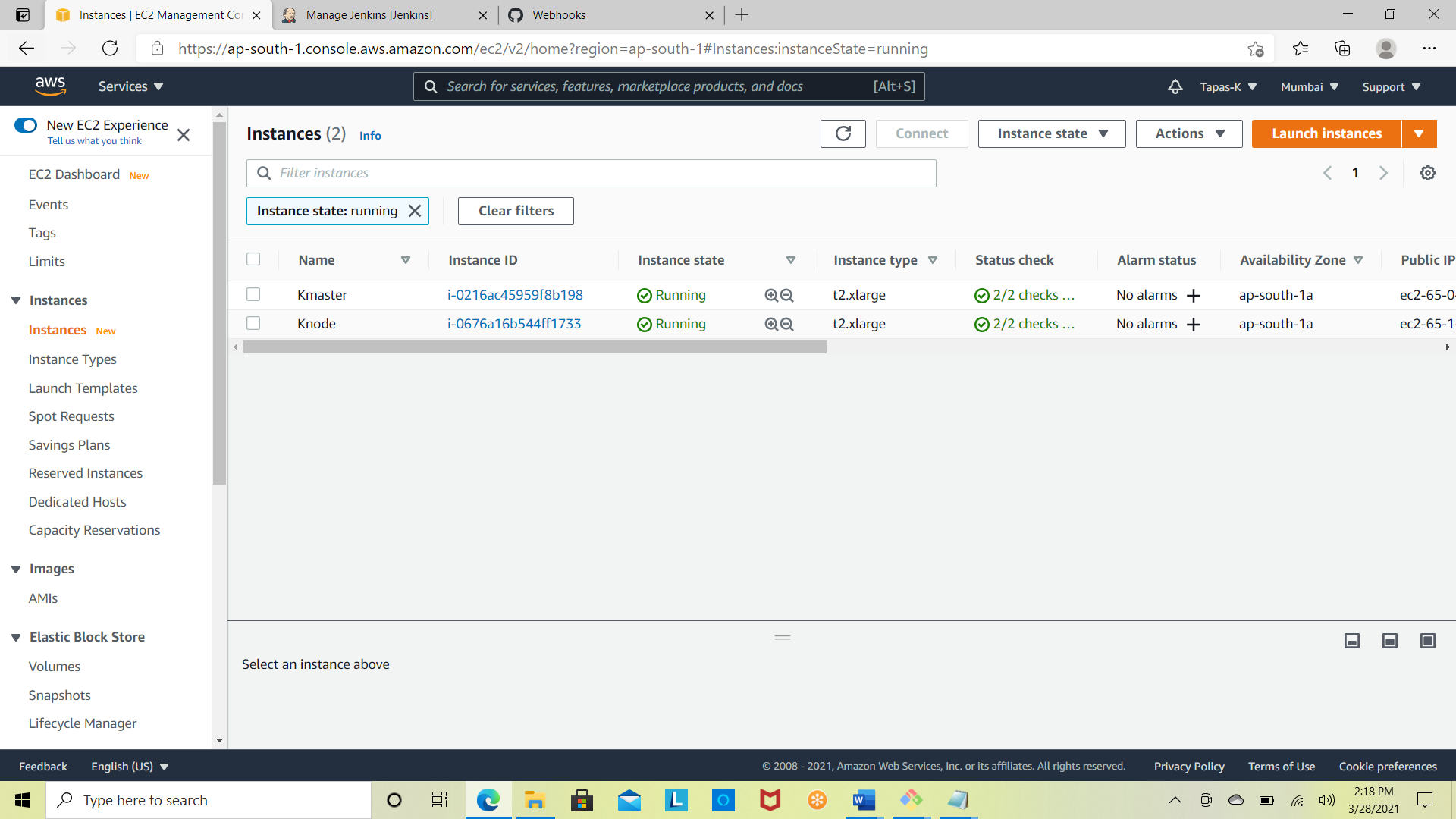
**Capstone Project – Tapaswini Kandala**

* **Executed the project with 2 t2.Xlarge instance.**



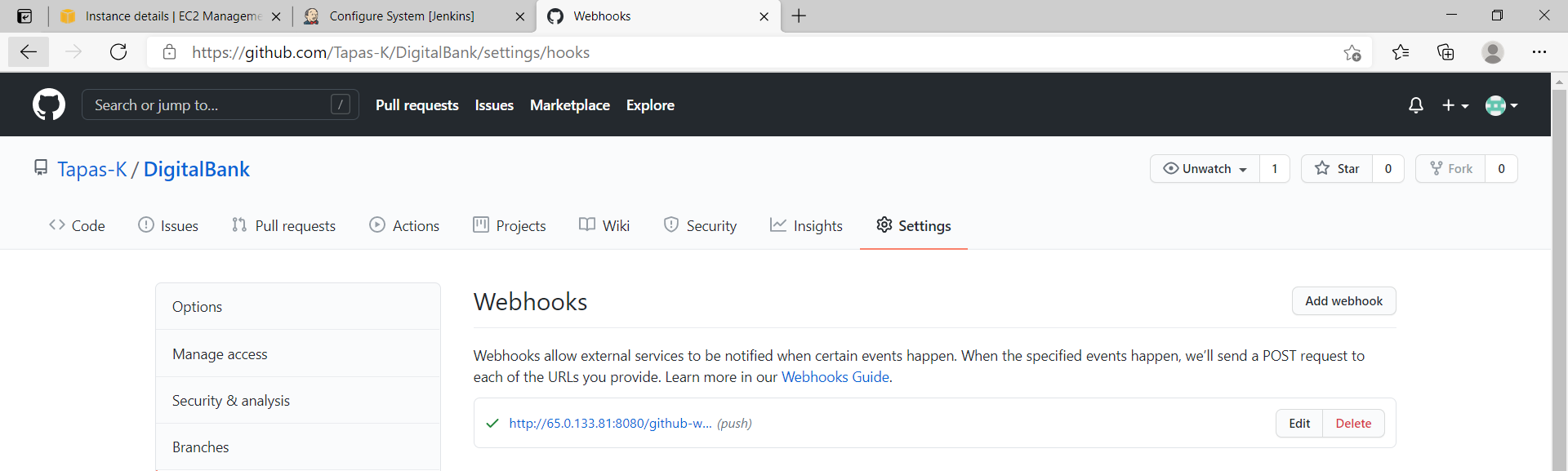
**STEP-1**

* Code has been migrated to GitHub

<https://github.com/Tapas-K/DigitalBank.git>

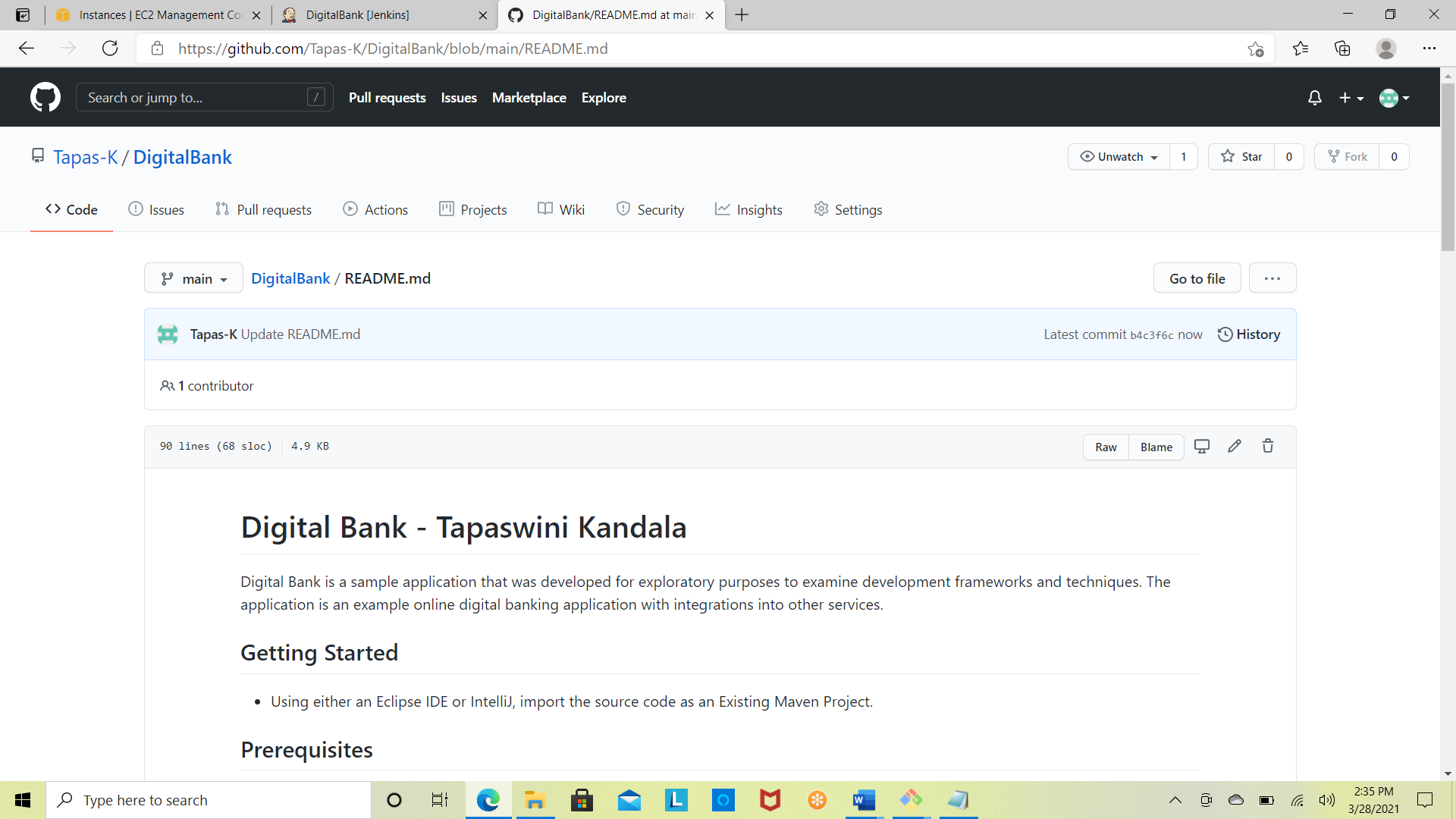
**STEP-2**

* Continuous Delivery has been implemented using Jenkins
* Pipeline Maven Integration Plugin installed
* Webhooks has been enabled
* GitHub hook trigger for GITScm polling enabled in Jenkins Pipeline





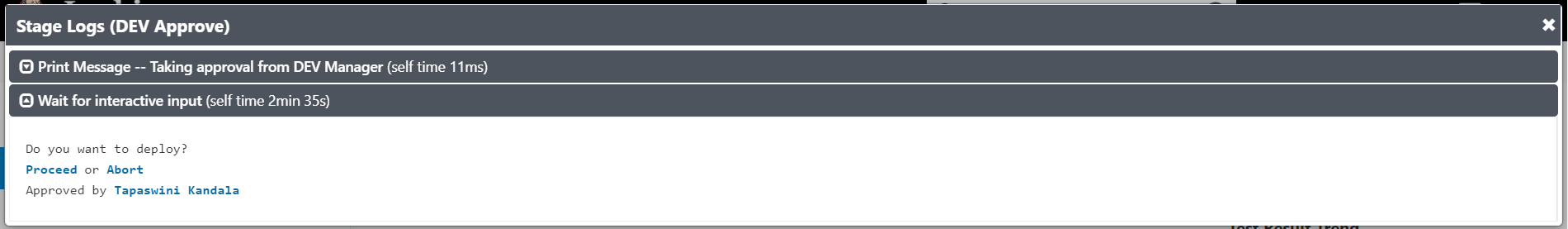
* On a Git push, Jenkins job has been triggered. This job contains source code building, testing and packaging stages as well.
* Updated the ReadMe file in Git repo directly and I could see the Jenkins Job triggered on Git push changes.

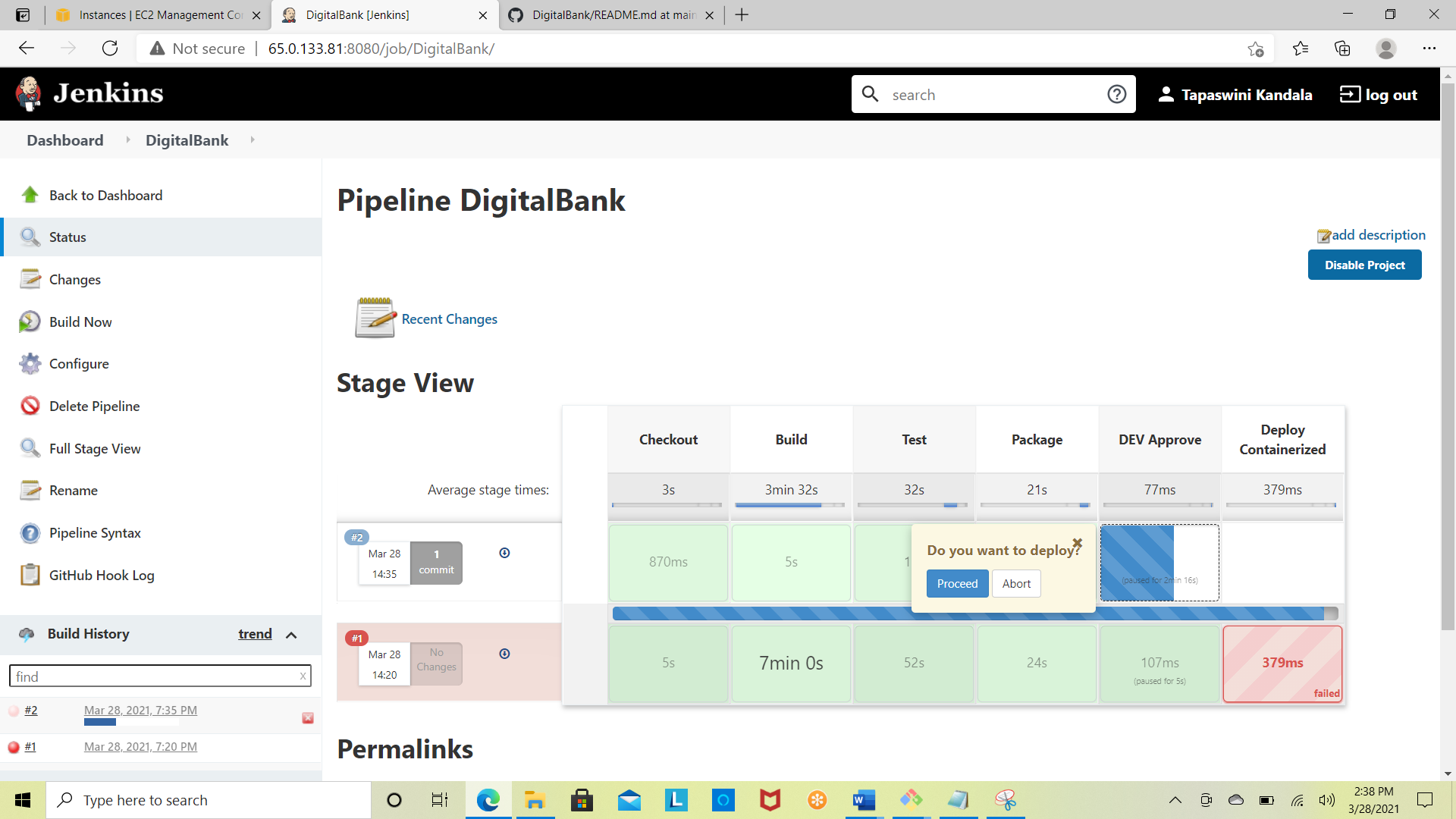


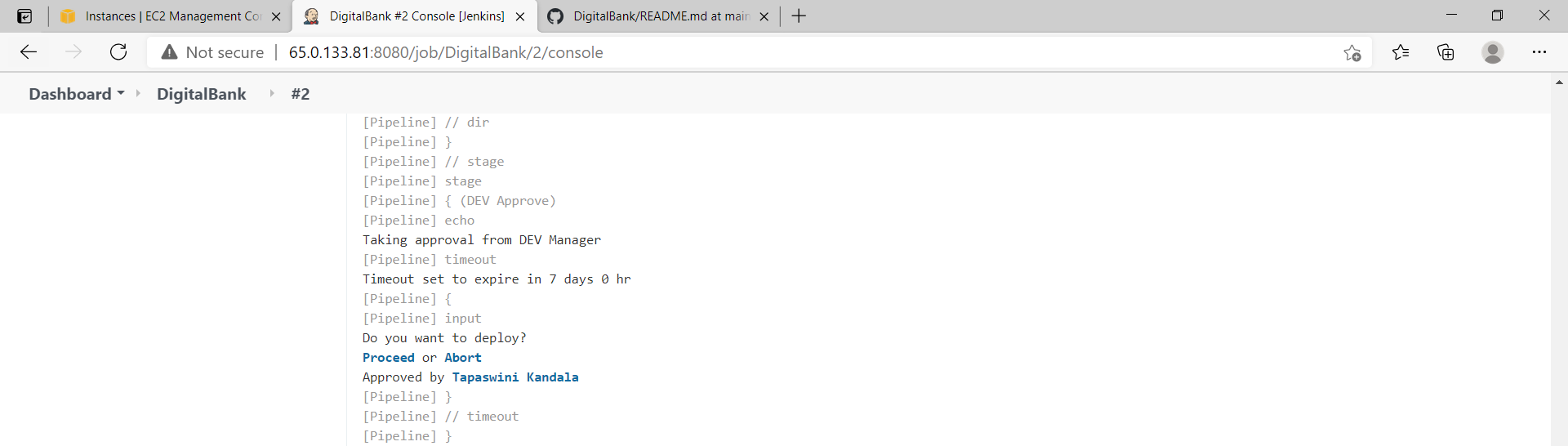


**STEP-3**

* Implemented a one-click deployment since this is a banking software the expectation is to ensure only a human can authorize the deployment of the software.
* Authorization has been given by Dev manager to proceed with the deployment







**STEP-4**

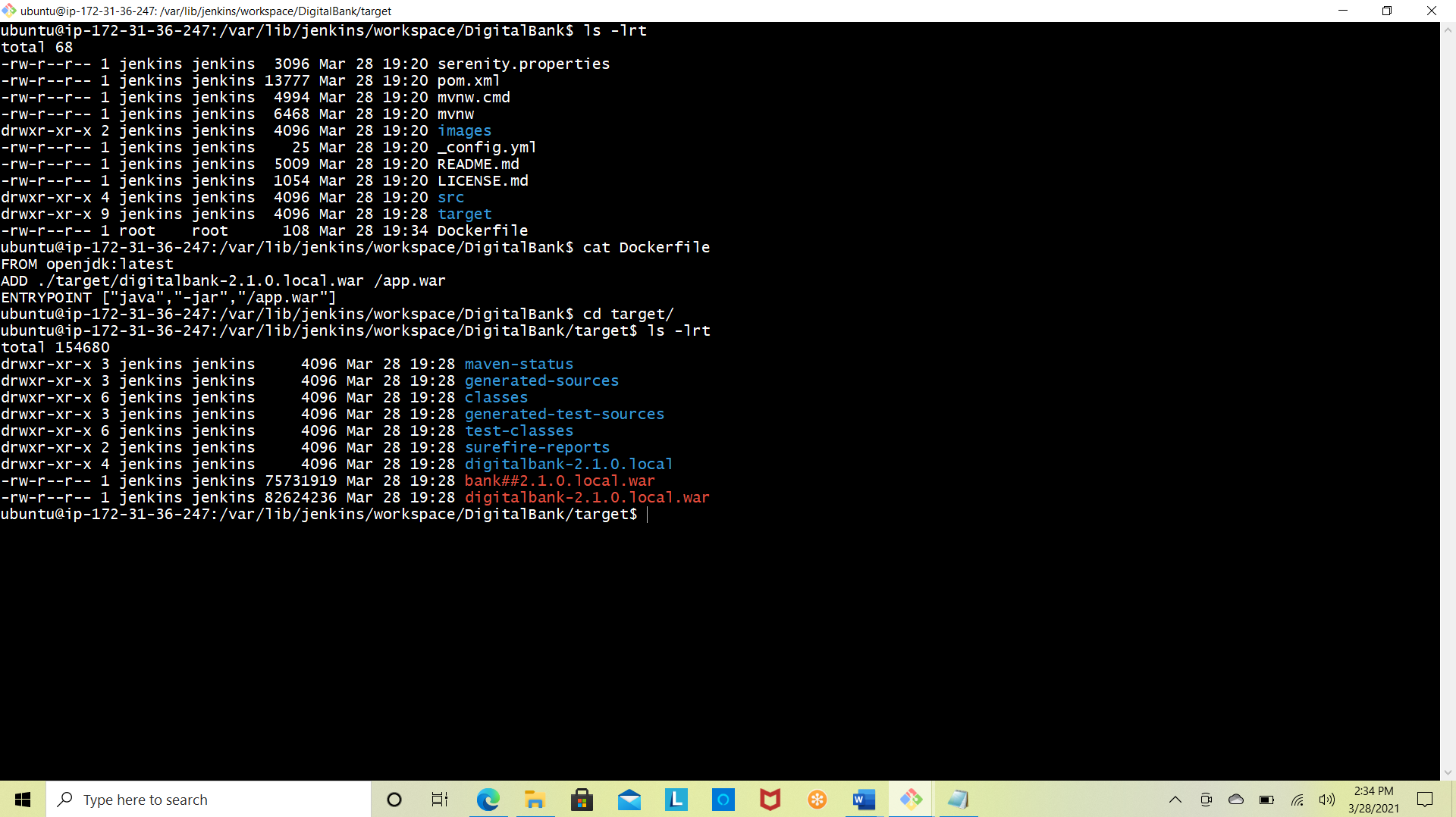
* The deployment package has been containerized in a Docker container.
* Install Docker
* sudo usermod -aG docker jenkins
* sudo visudo
* jenkins ALL=(ALL) NOPASSWD: ALL

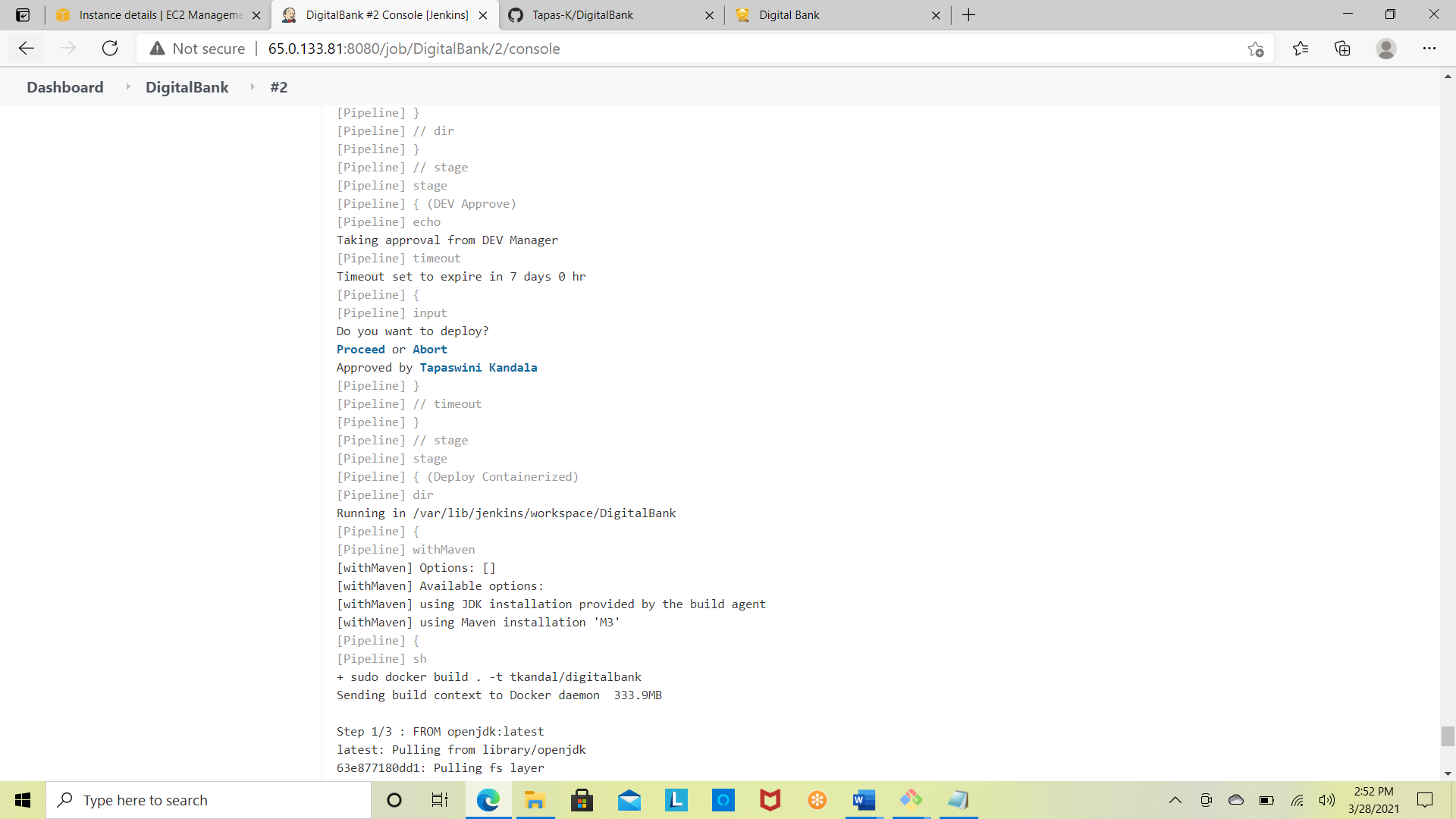
**Dockerfile** has been placed under/var/lib/jenkins/workspace/DigitalBank

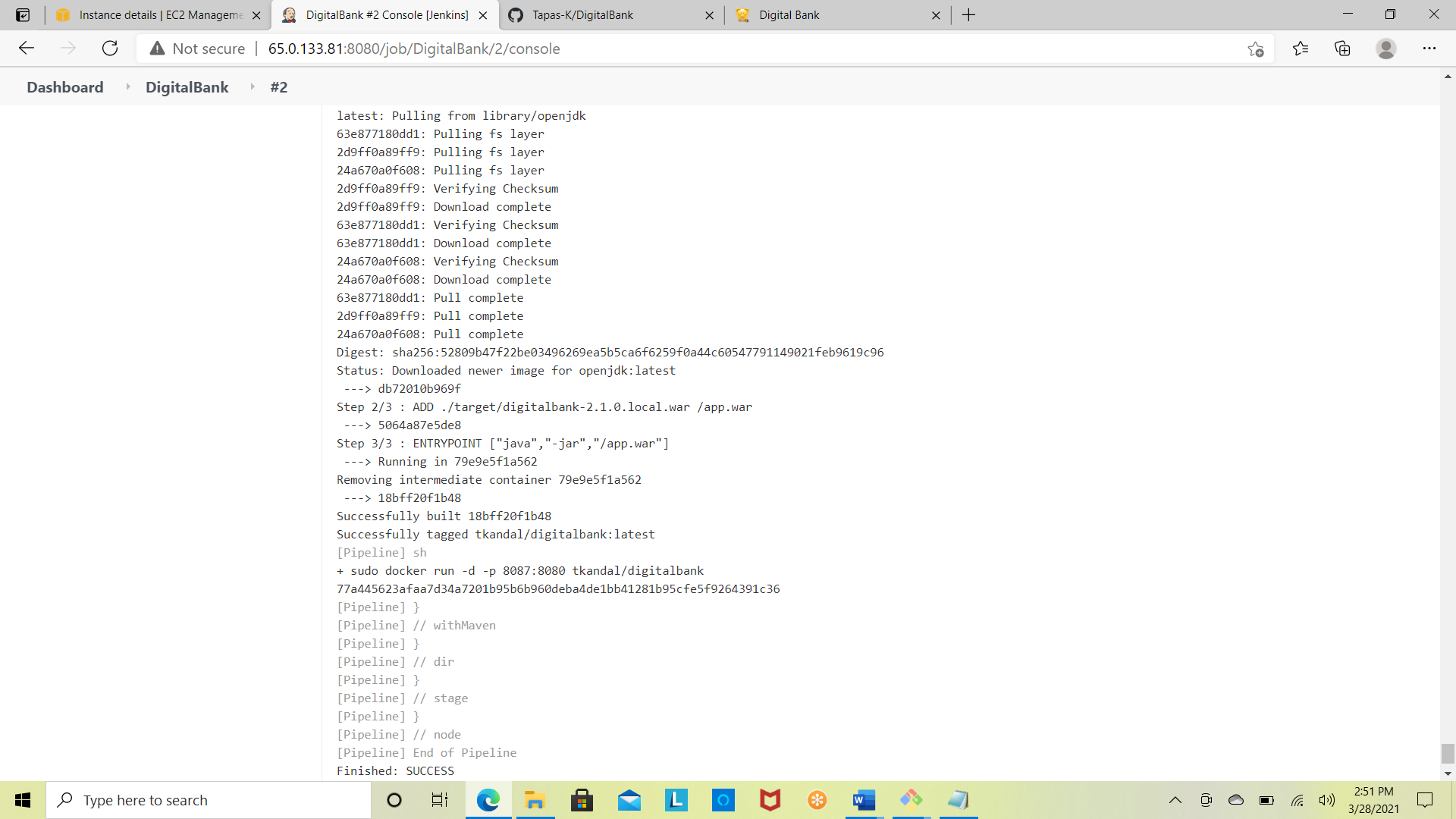
FROM openjdk:latest

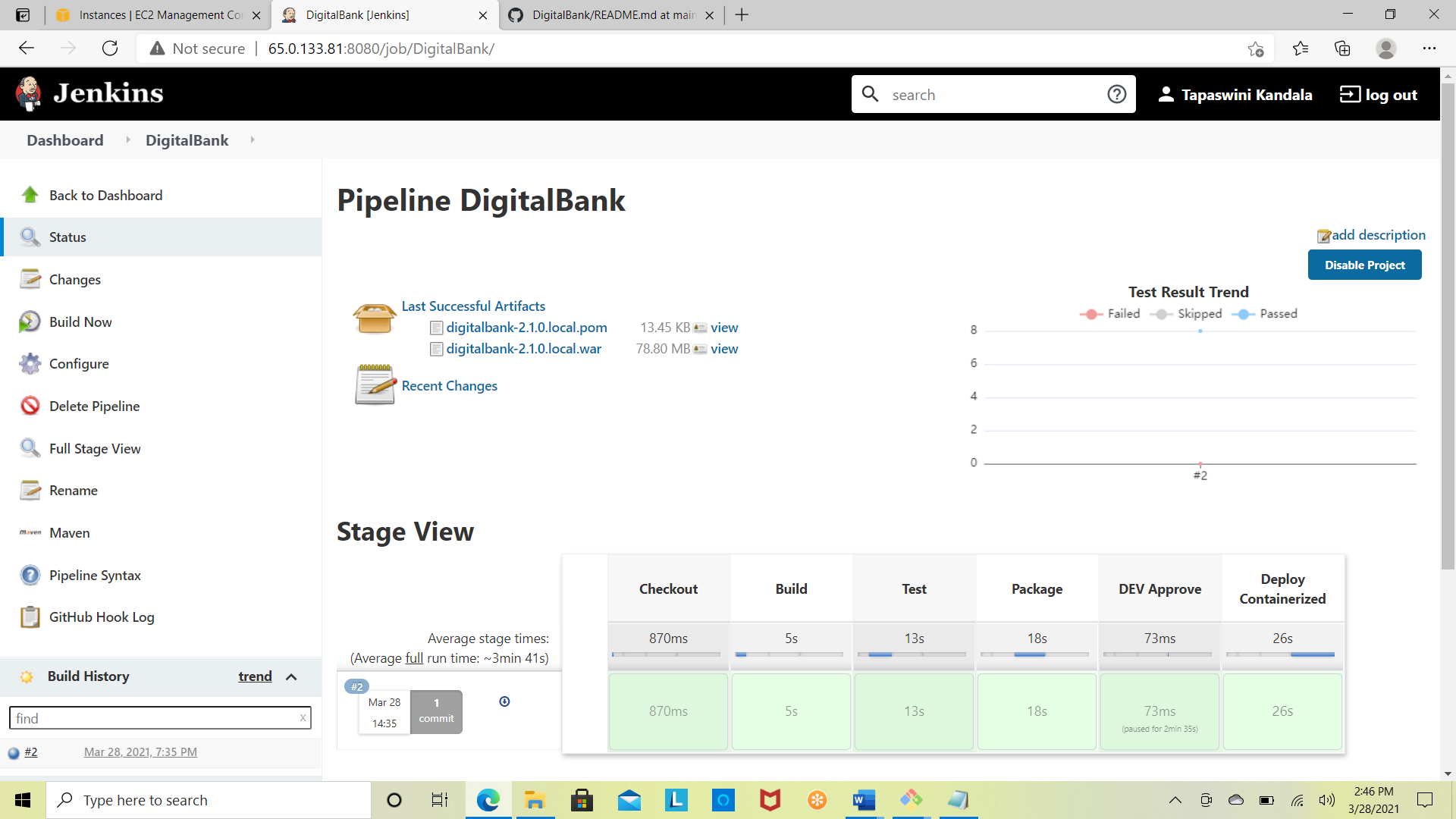
ADD ./target/digitalbank-2.1.0.local.war /app.war

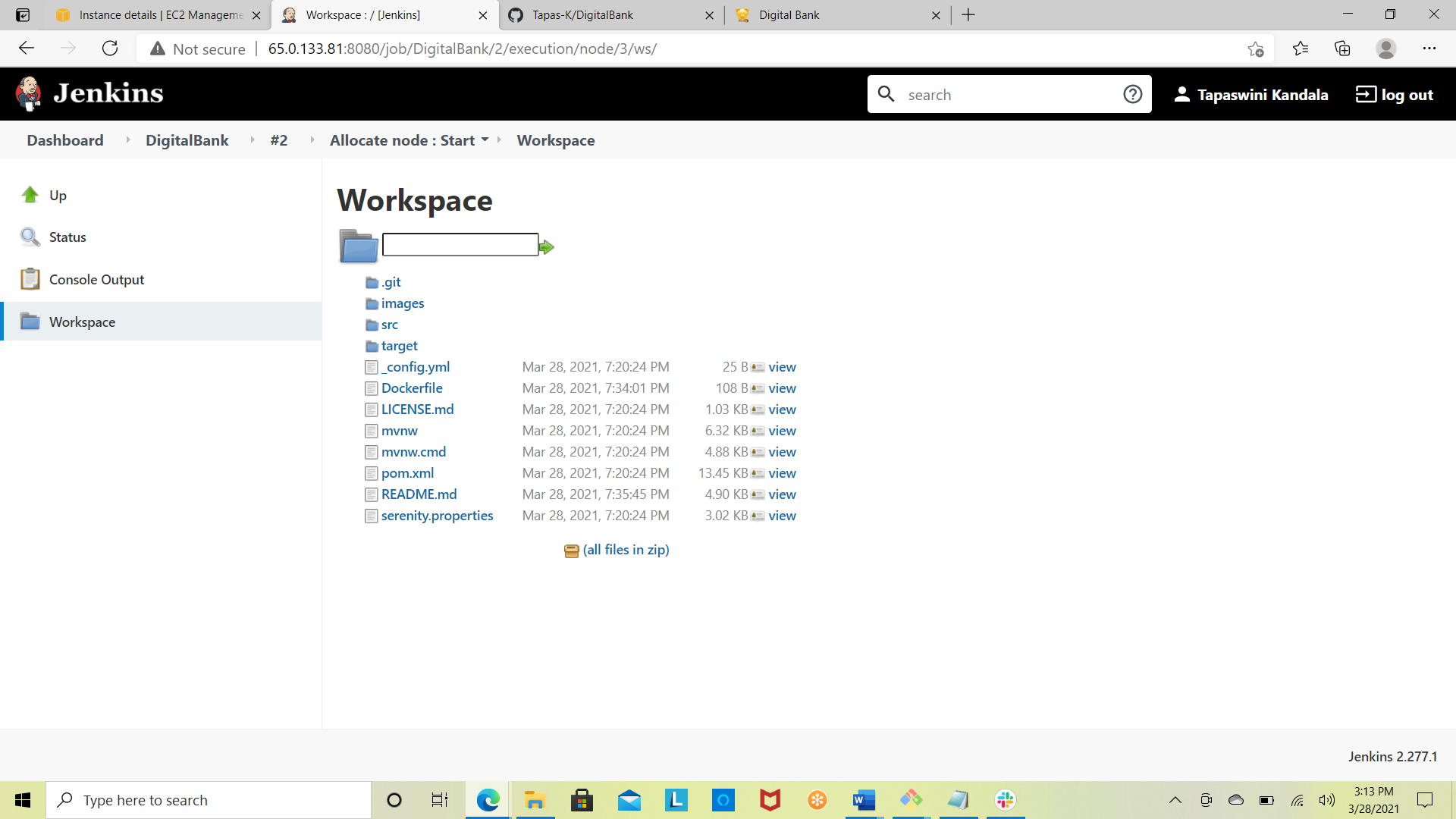
ENTRYPOINT ["java","-jar","/app.war"]



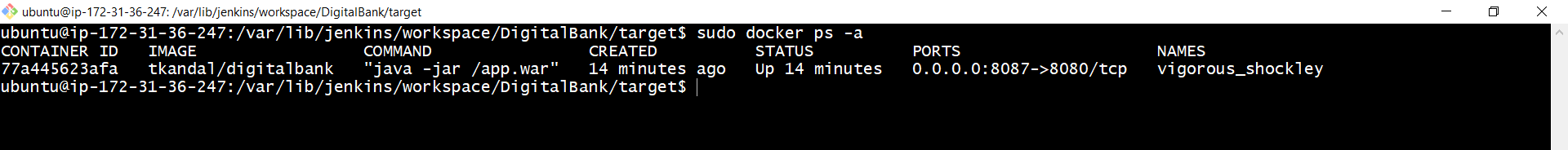


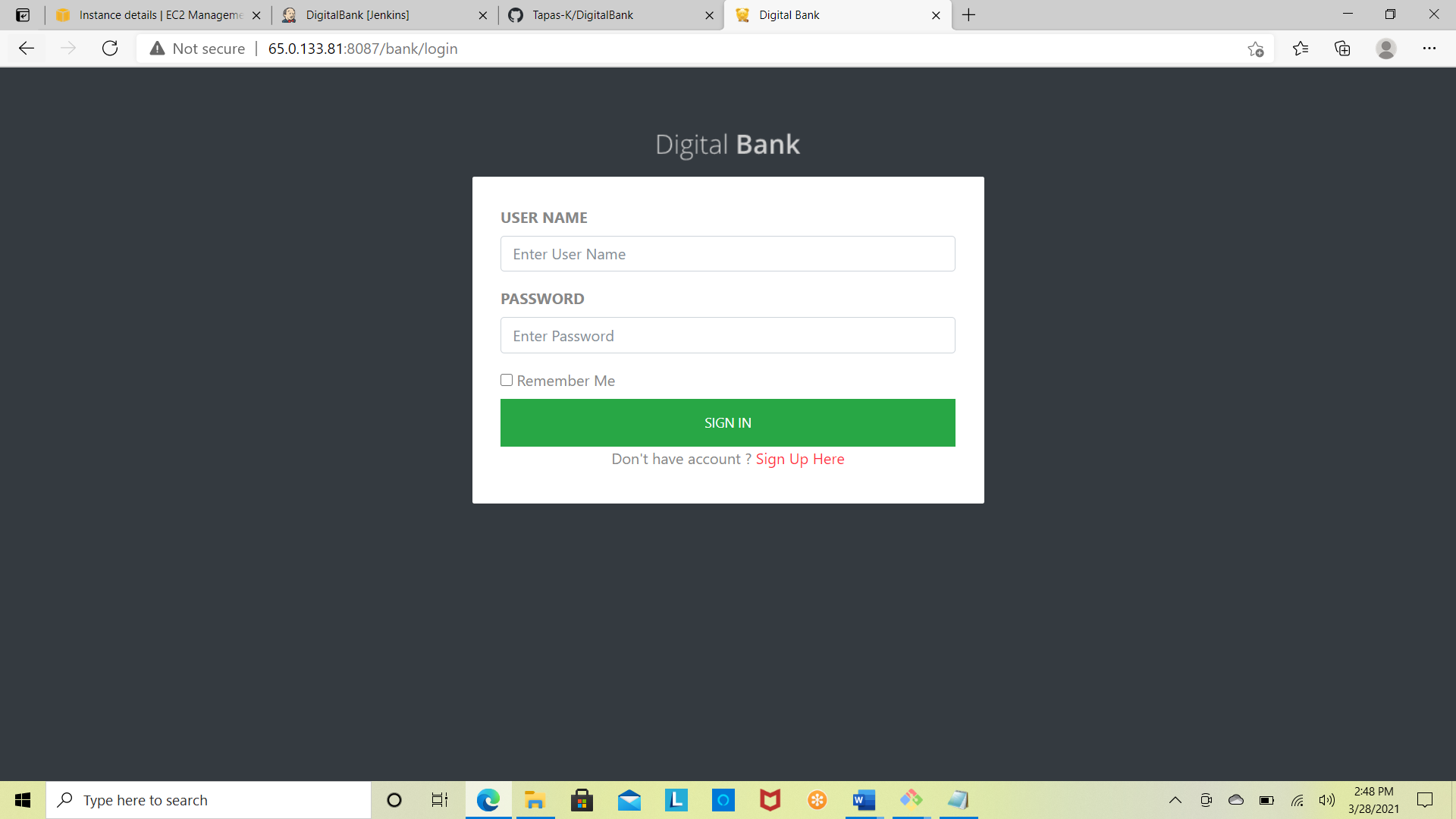


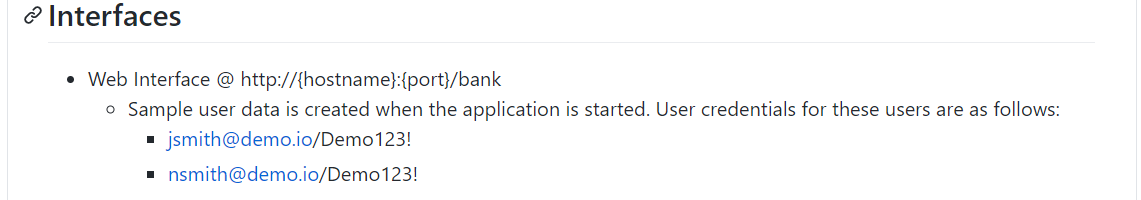


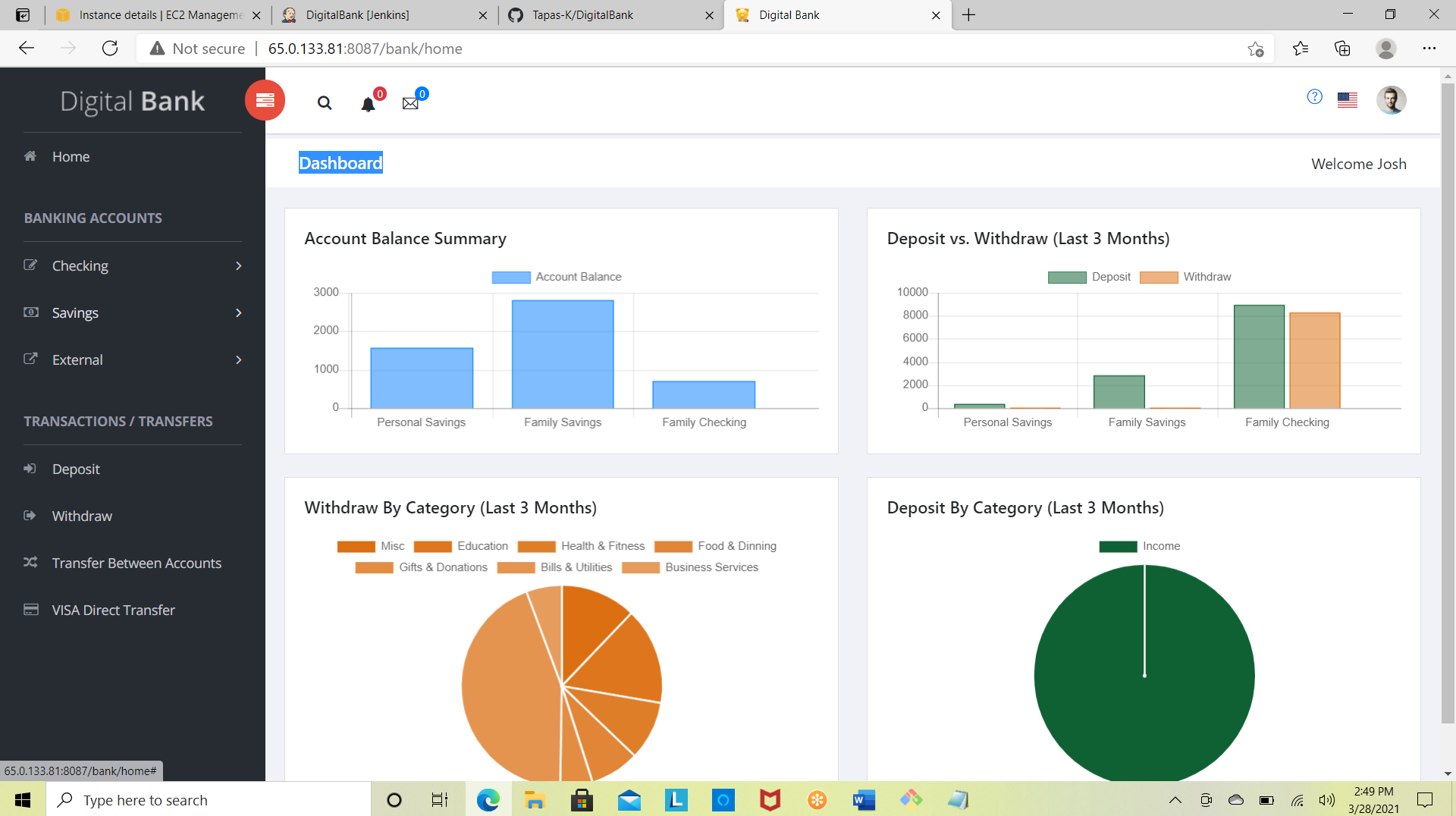


* Post the docker container deployment, was able to successfully access the application with the port 8087 in the below screenshot with the credentials that have been shared in the README.md file.
* Checked the docker container is running on the server





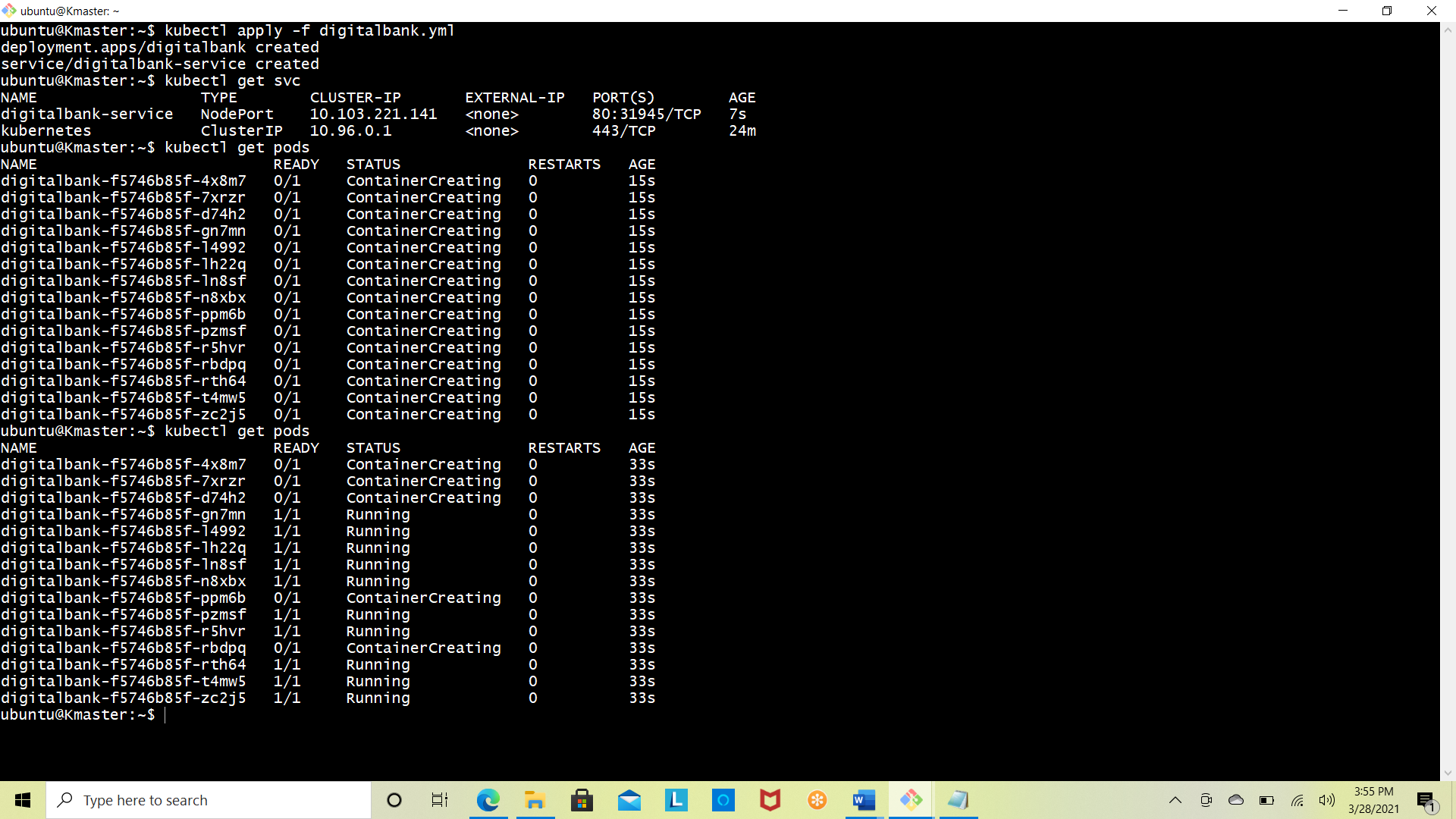


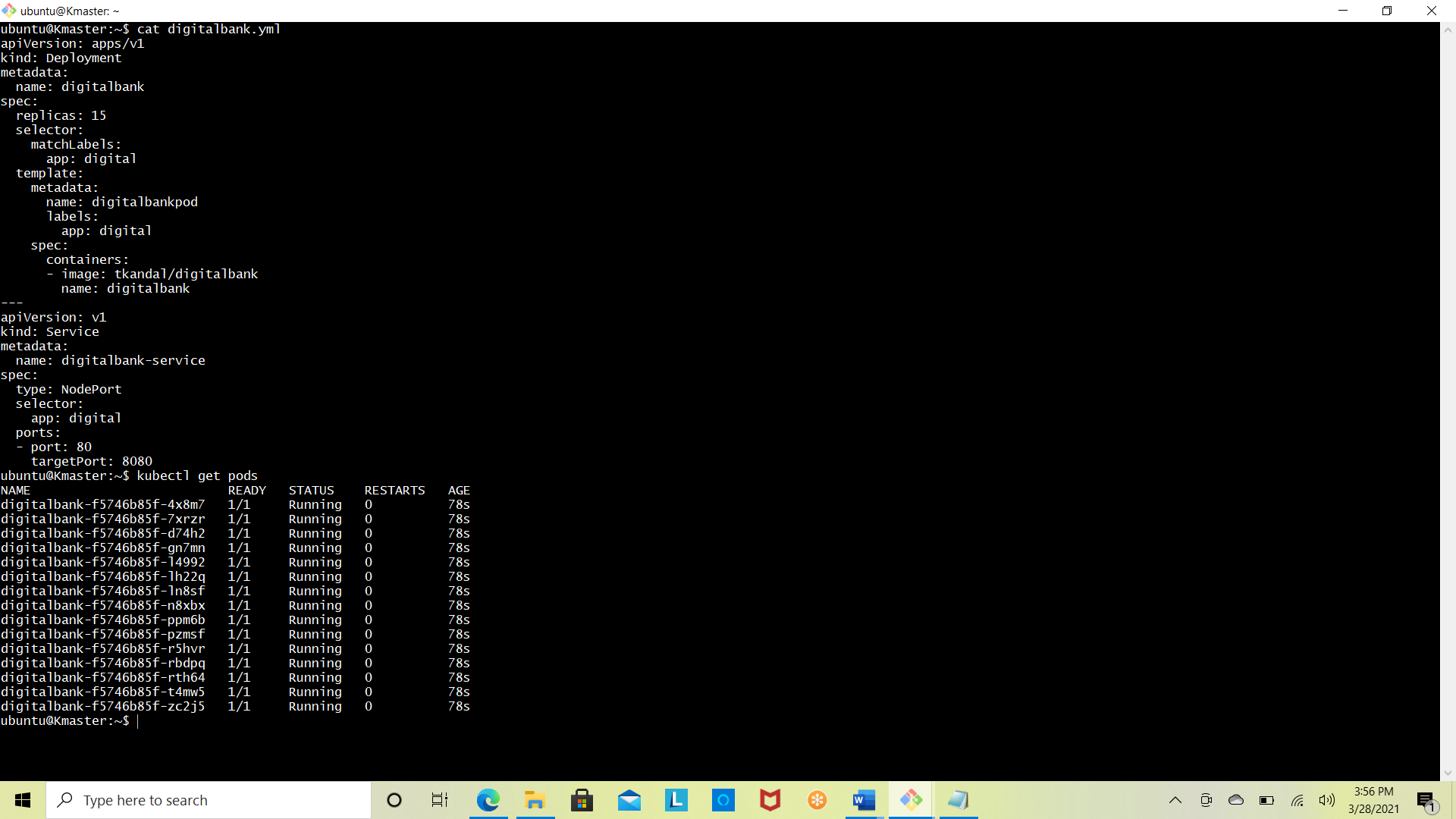


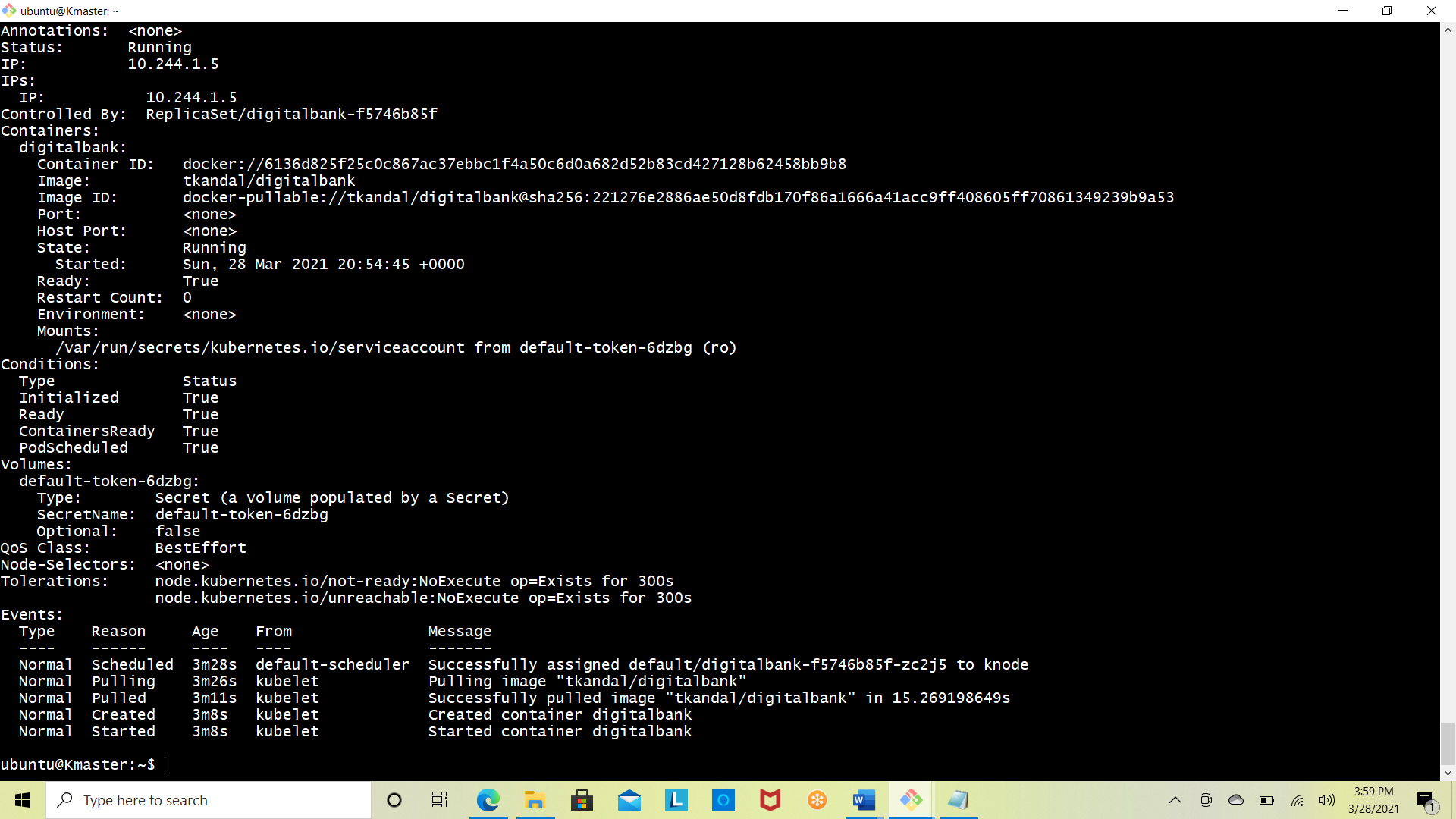
**STEP-5**

* Deployed the containerized application in AWS – 15 instances of the application are running all the time to ensure the availability using Kubernetes Cluster.
* Installed Kubernetes Cluster in Kmaster & Knode
* Both of these nodes are running on t2.Xlarge instances to avoid resource issue between the Kmaster & Knode.

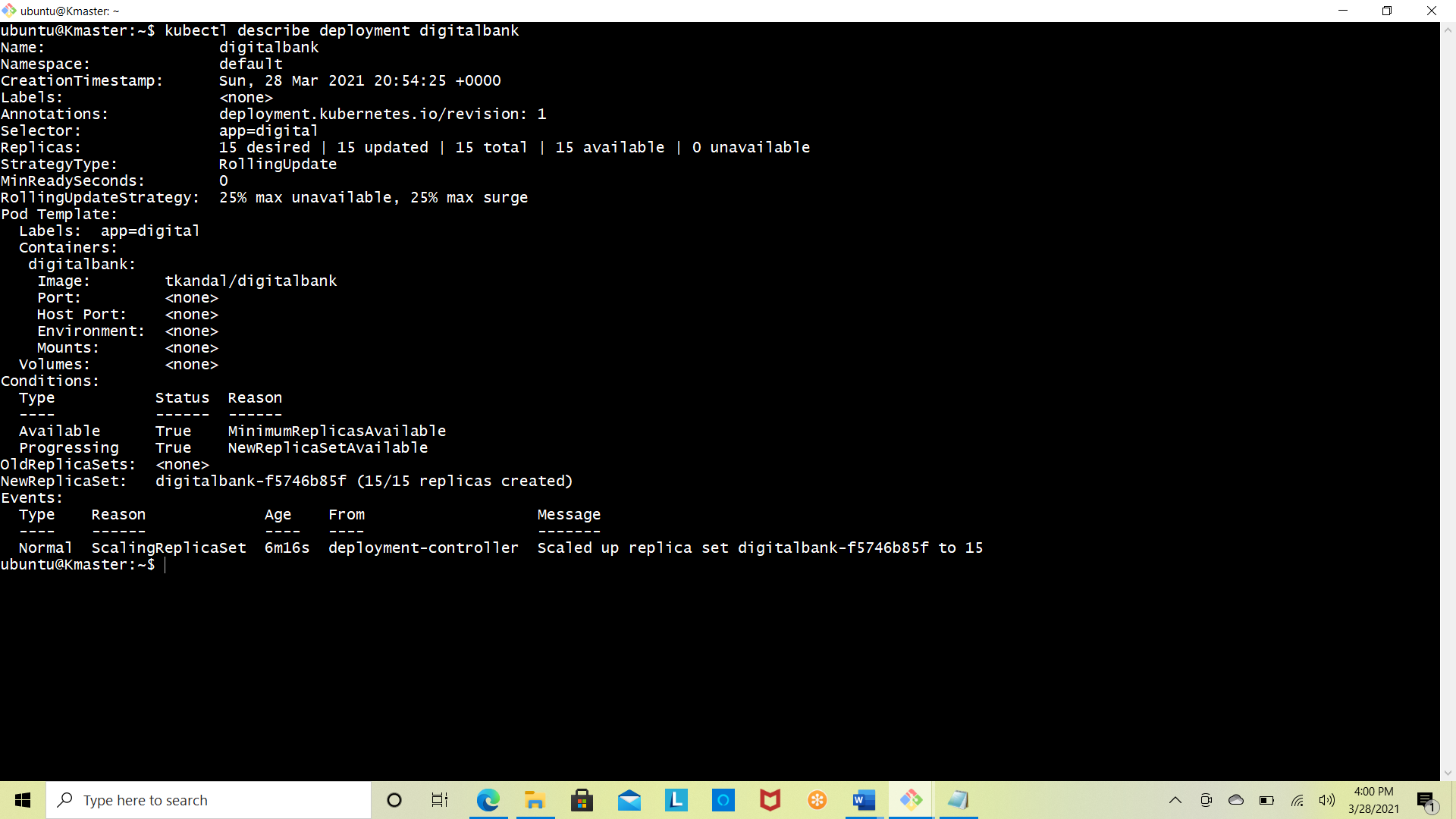
****



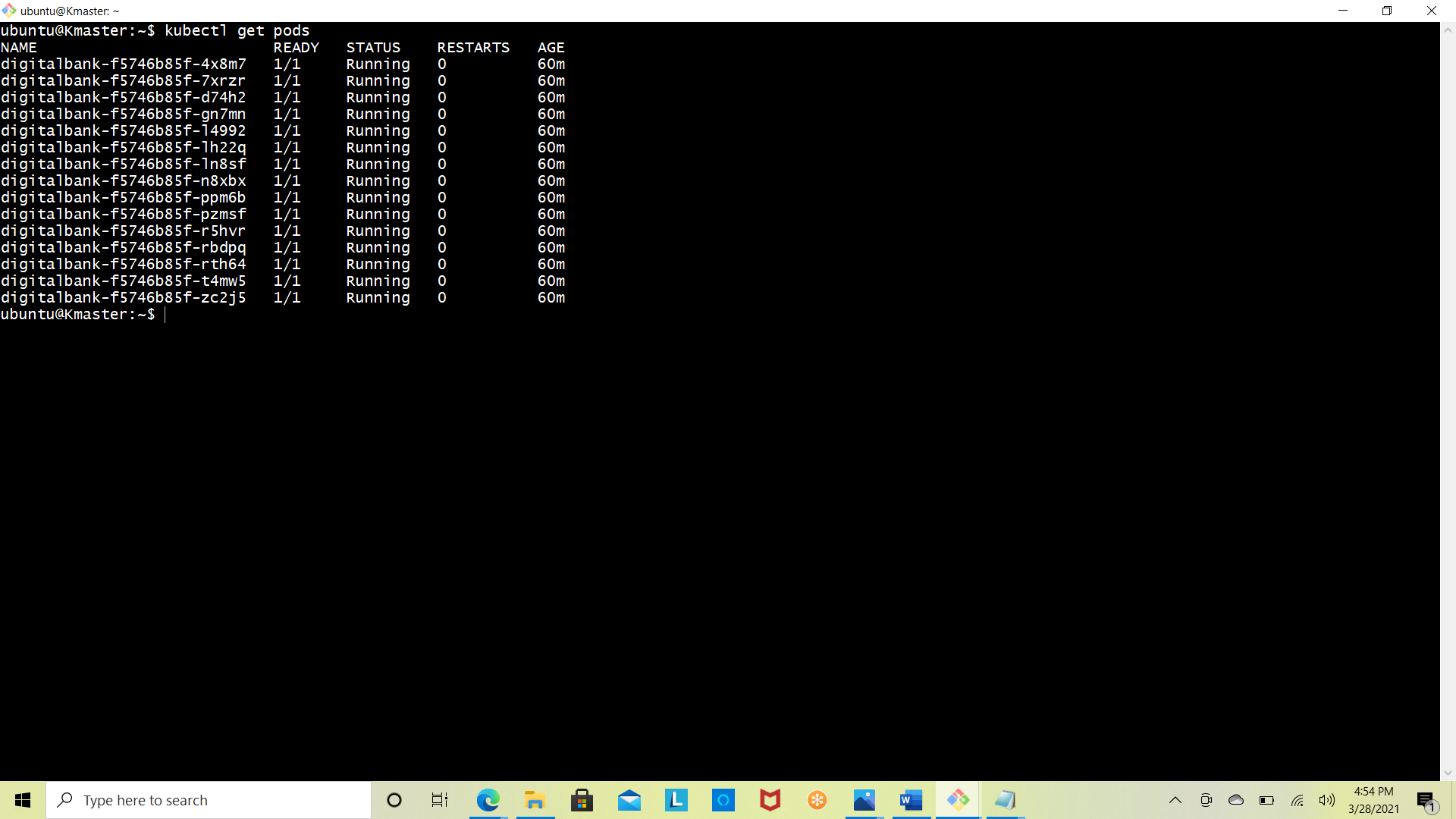




****

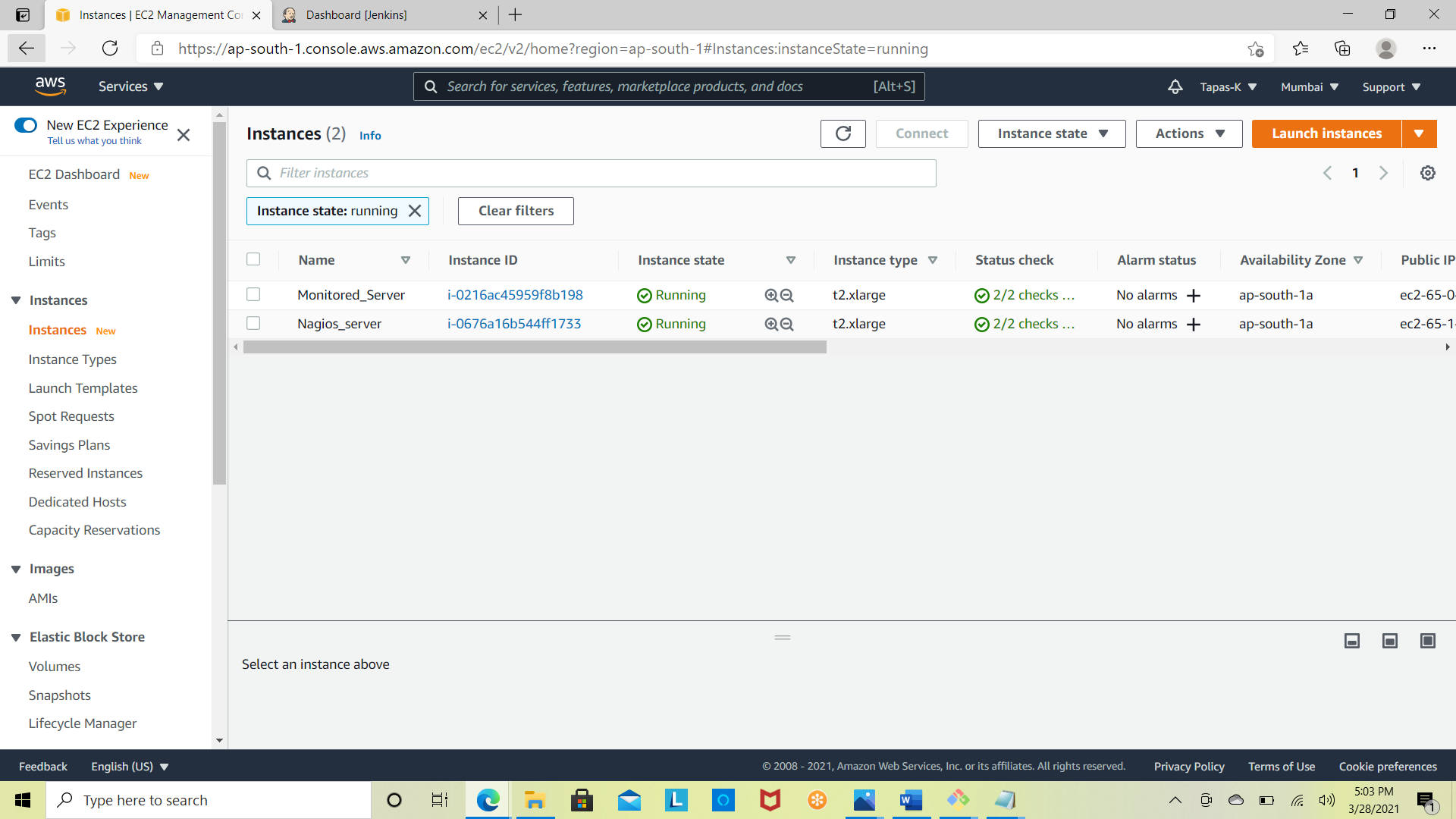


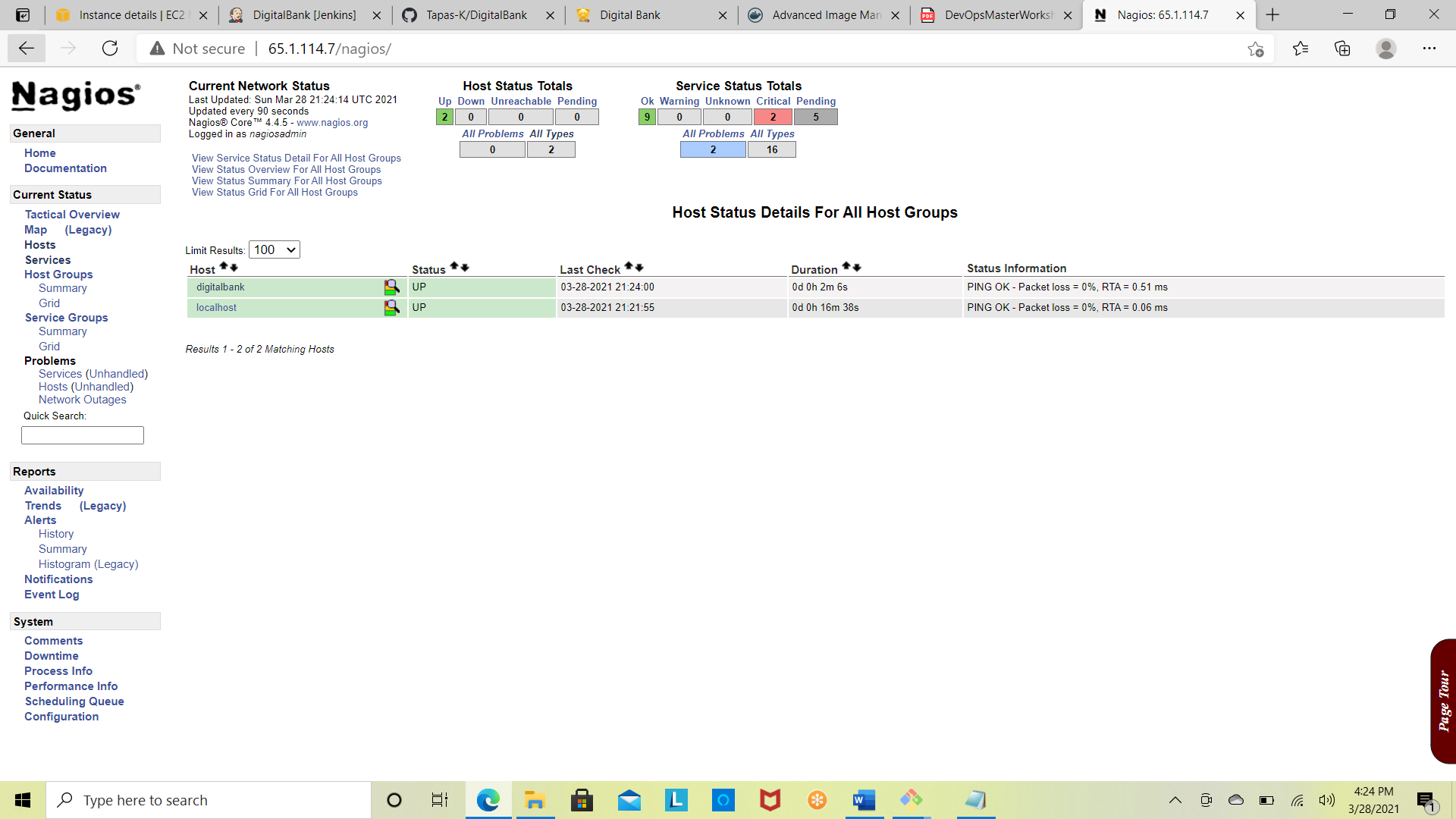
* **The pods are stable enough as we can see ALL the 15 PODS are up and running fine from the time of creation.**

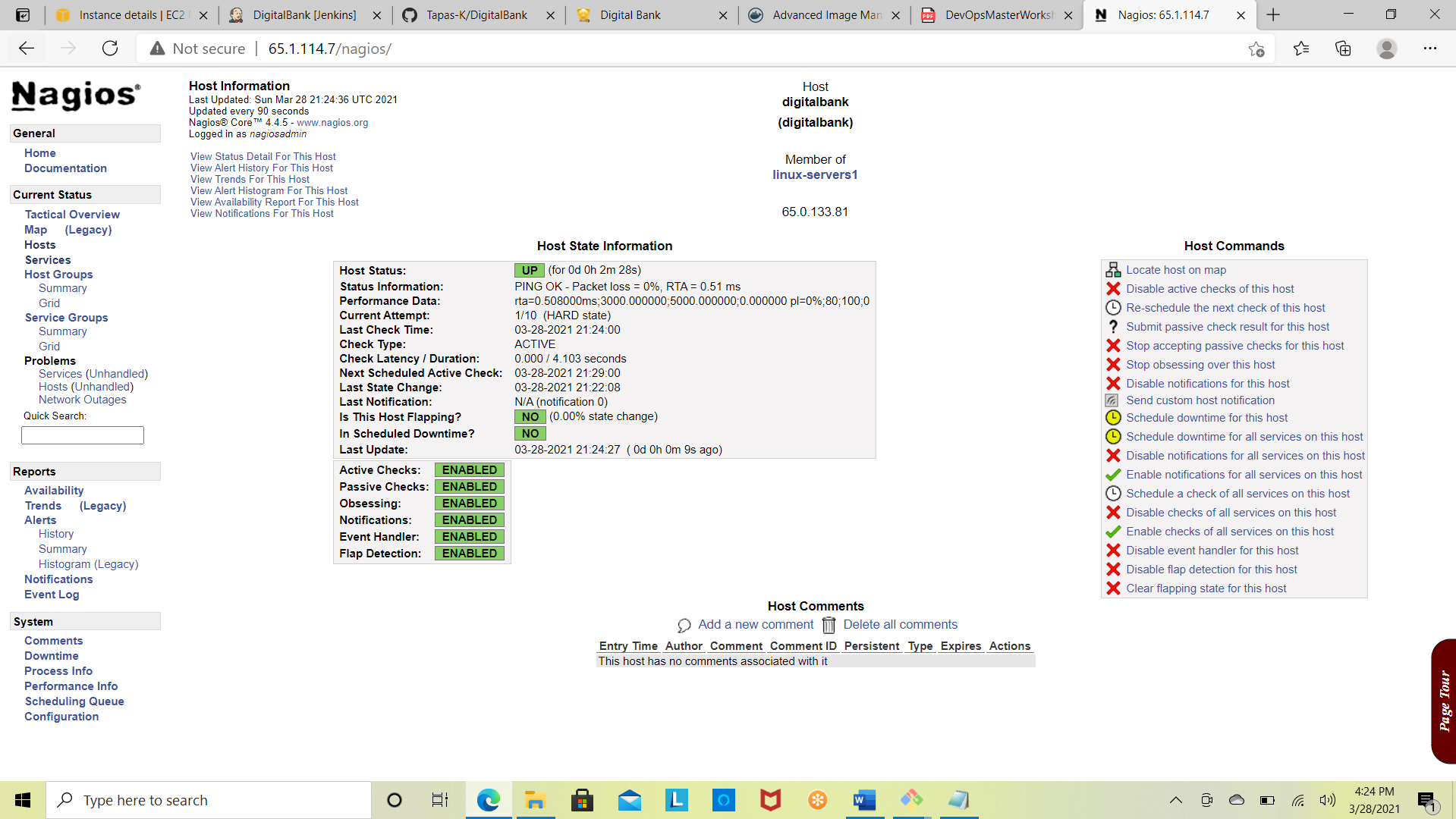


**STEP-6**

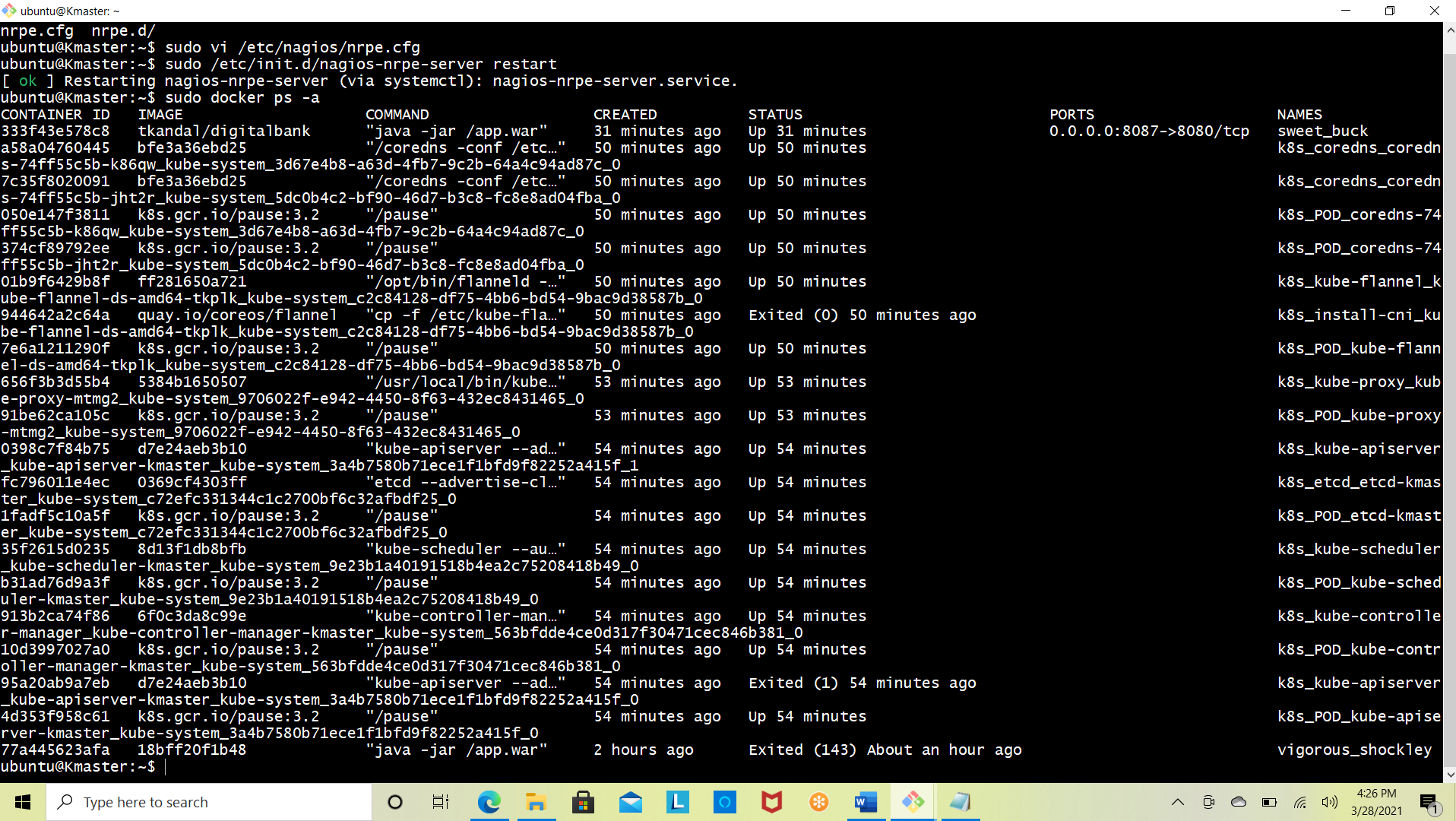
* **Successfully built a monitoring mechanism to monitor the Digital Bank host systems and services.**

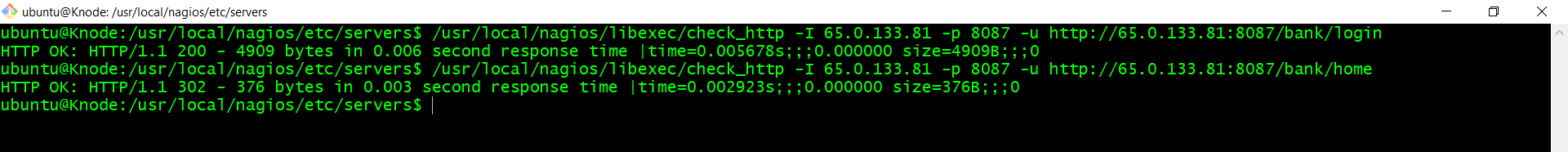


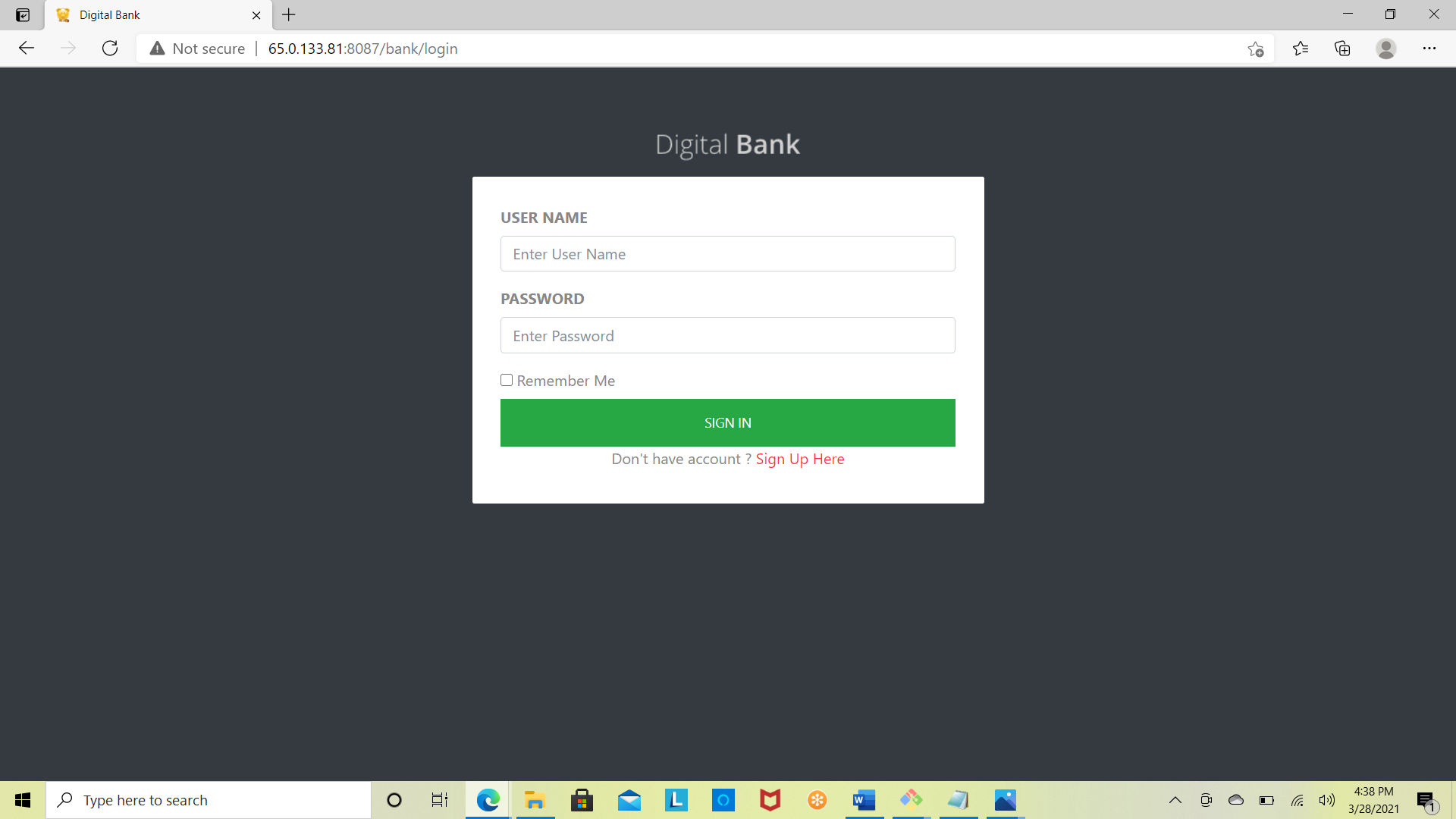


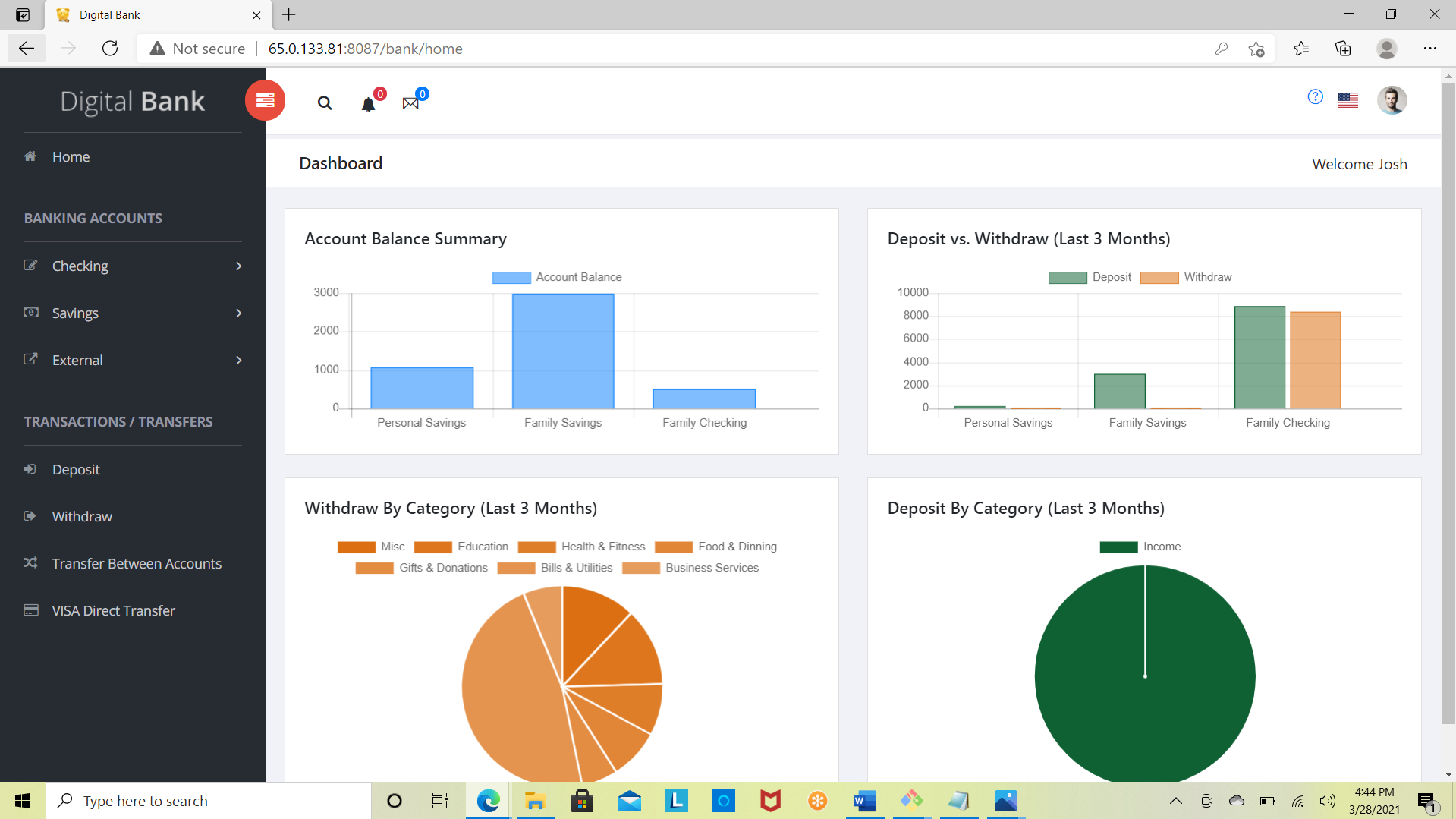


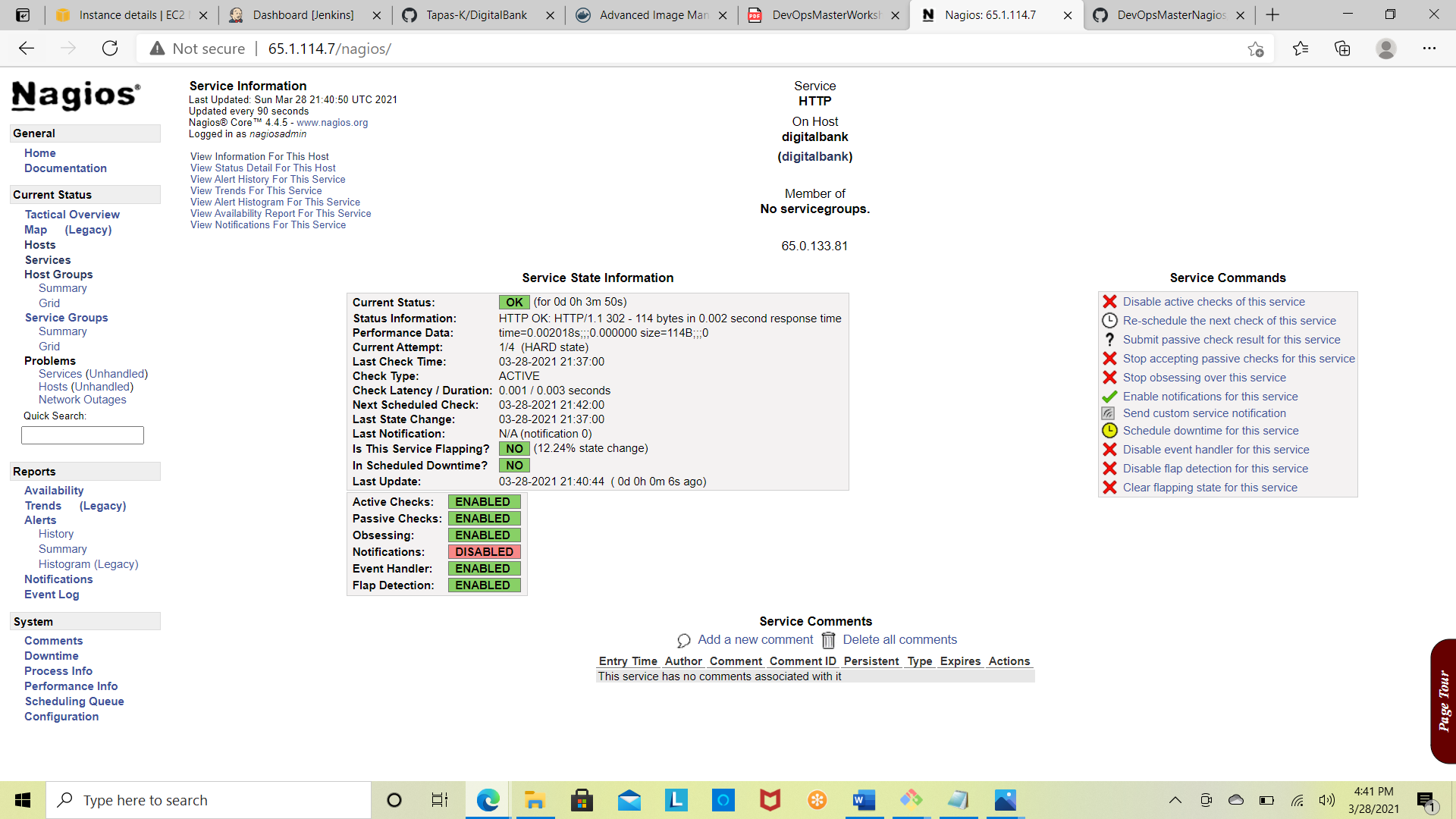
****

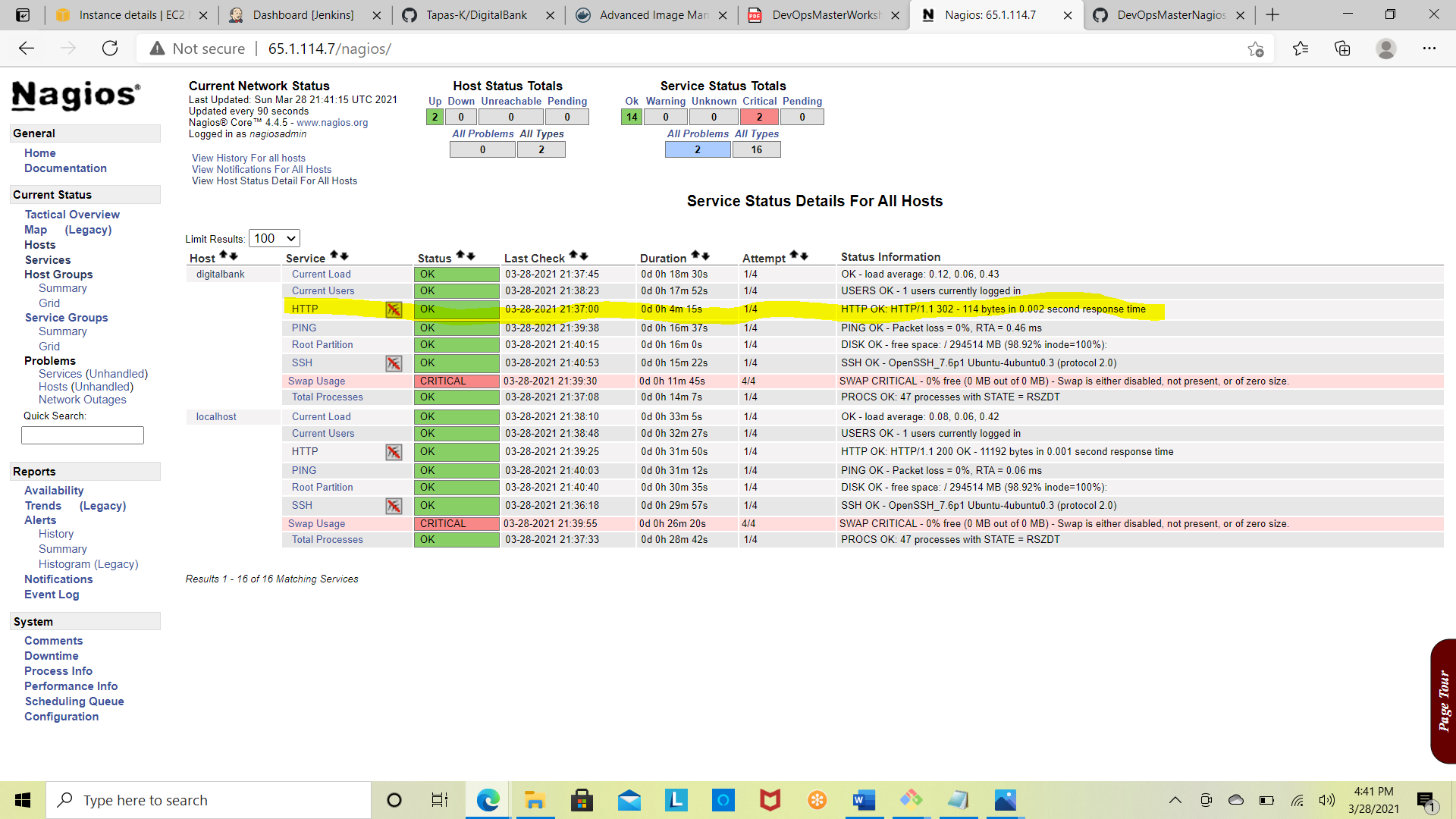








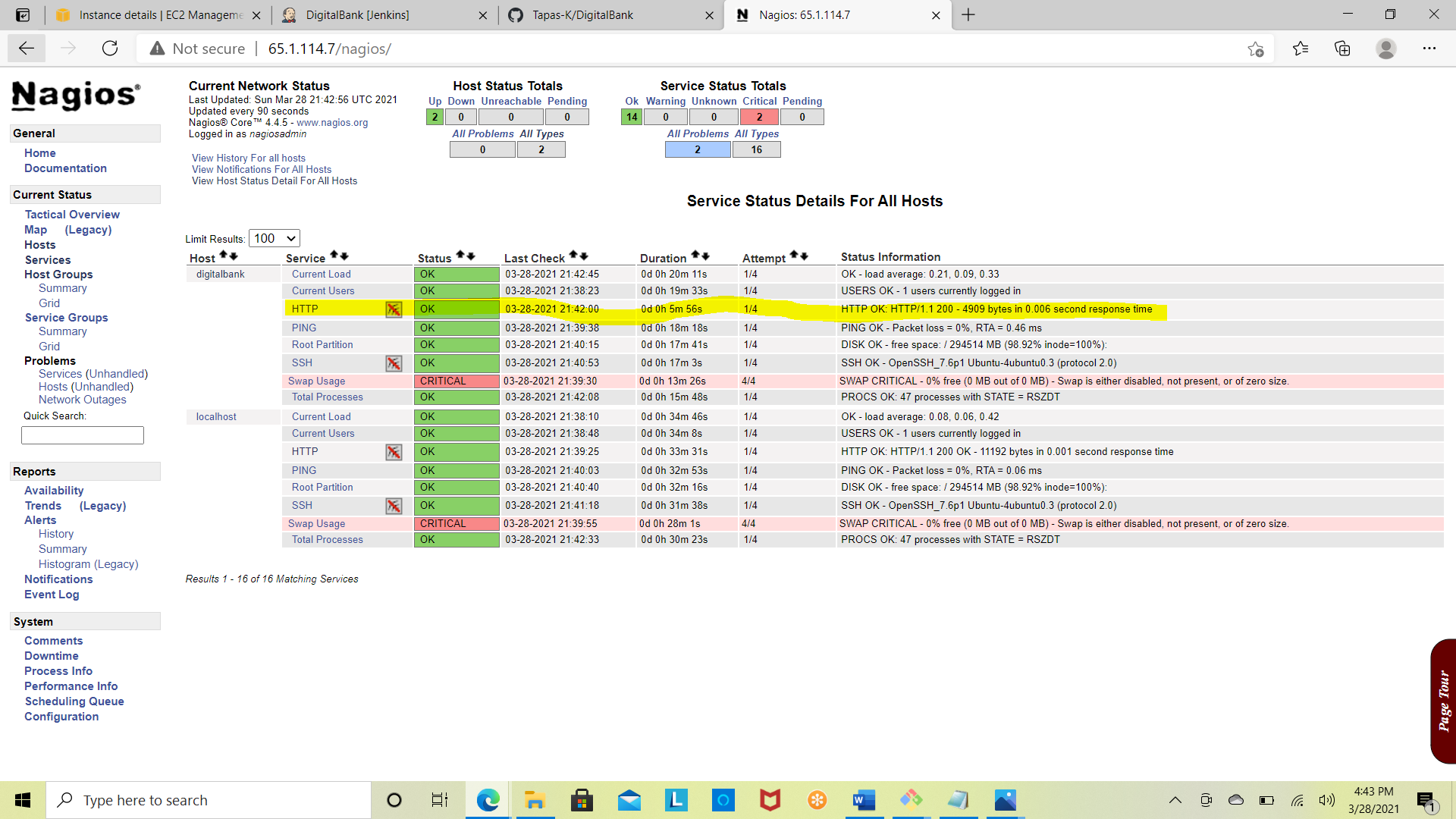


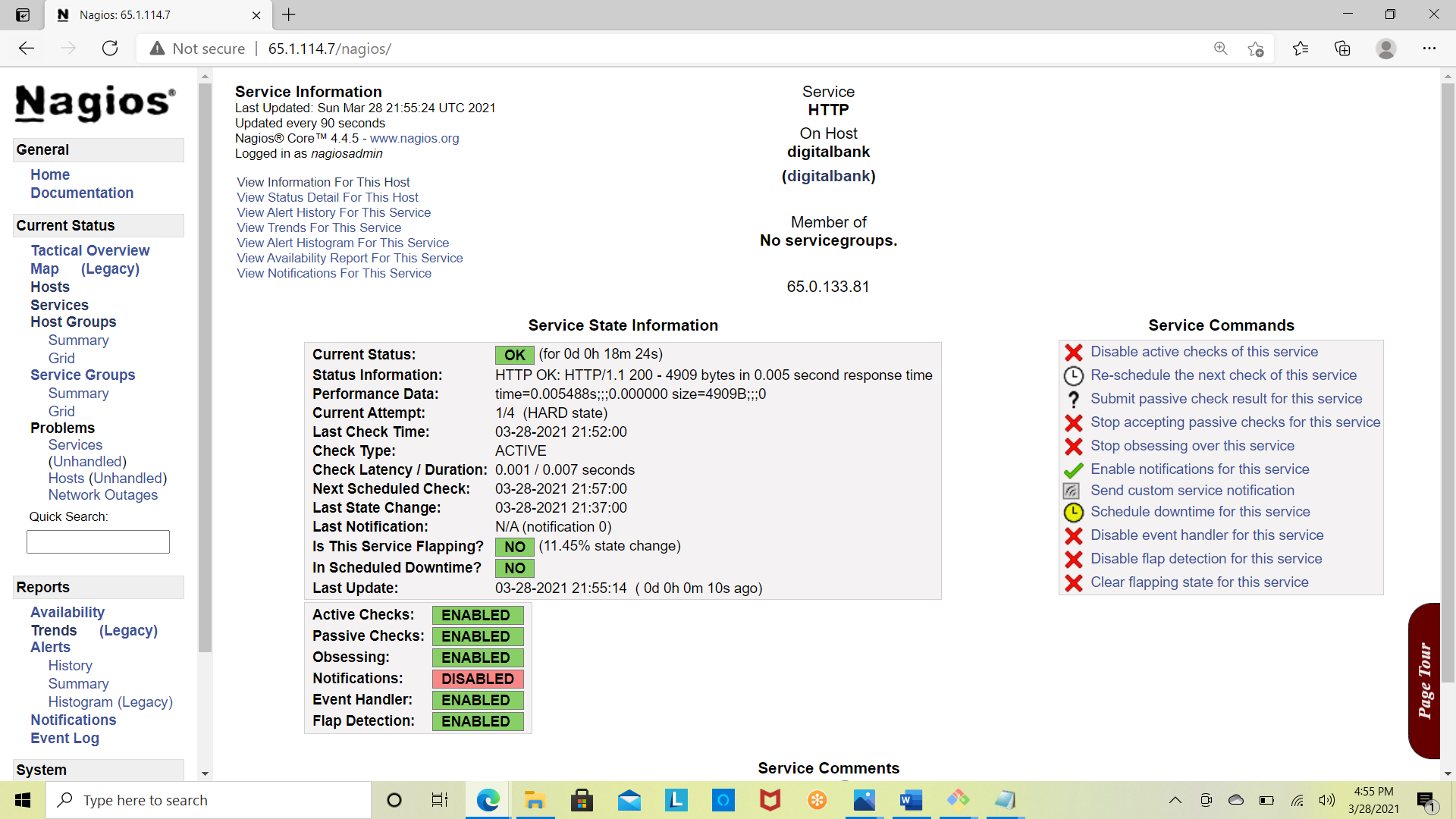


**Observations:**

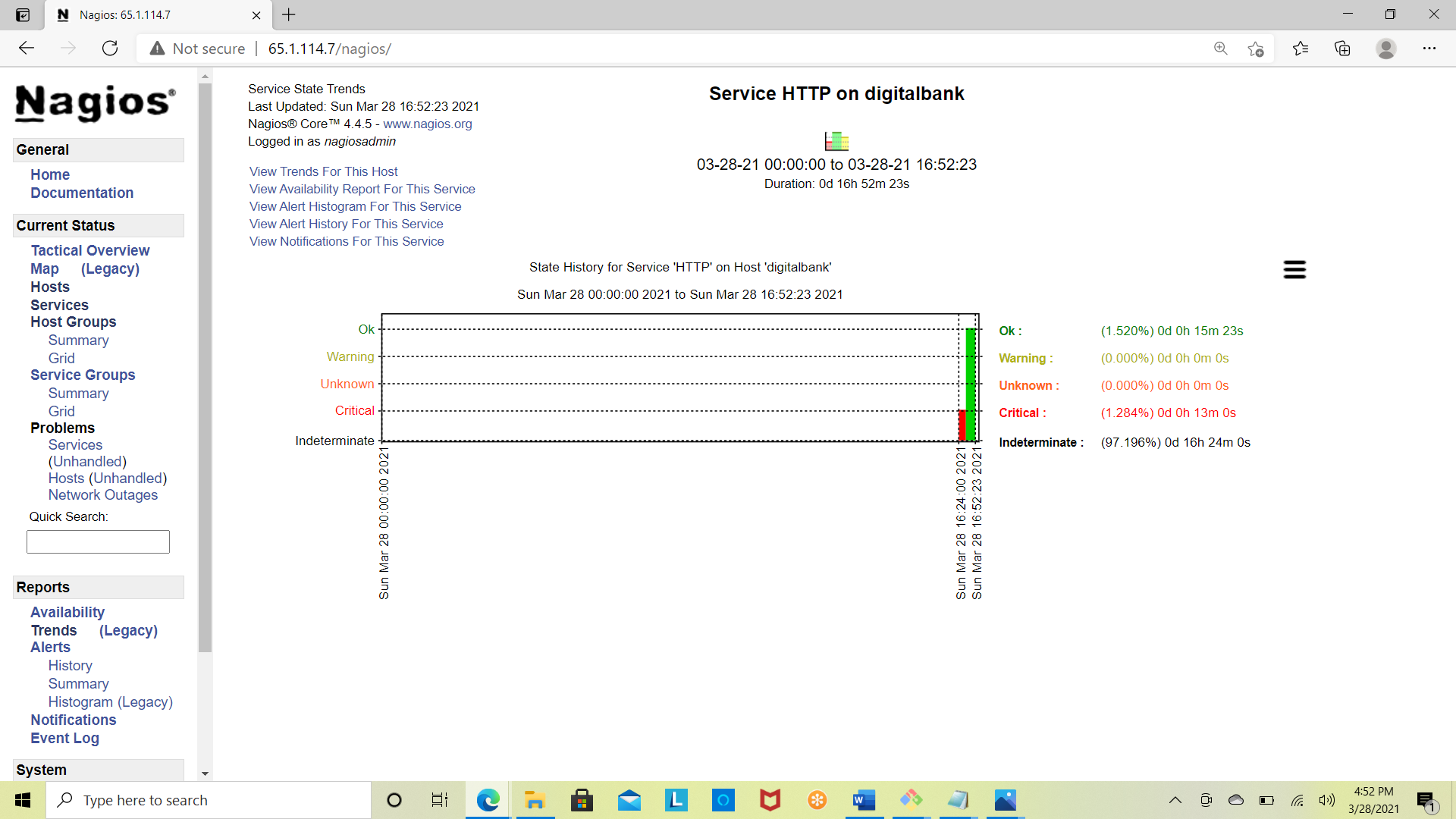
* **Http 302 code typically states redirection of the page when tested with** http://<Ipaddress>:8087/bank/home
* **Http 200 code reflects when the application is up and able to hit the page**

http://<Ipaddress>:8087/bank/login





**Nagios Digital Bank Service Monitoring Report**



**Nagios Digital Bank Host Monitoring Report**

