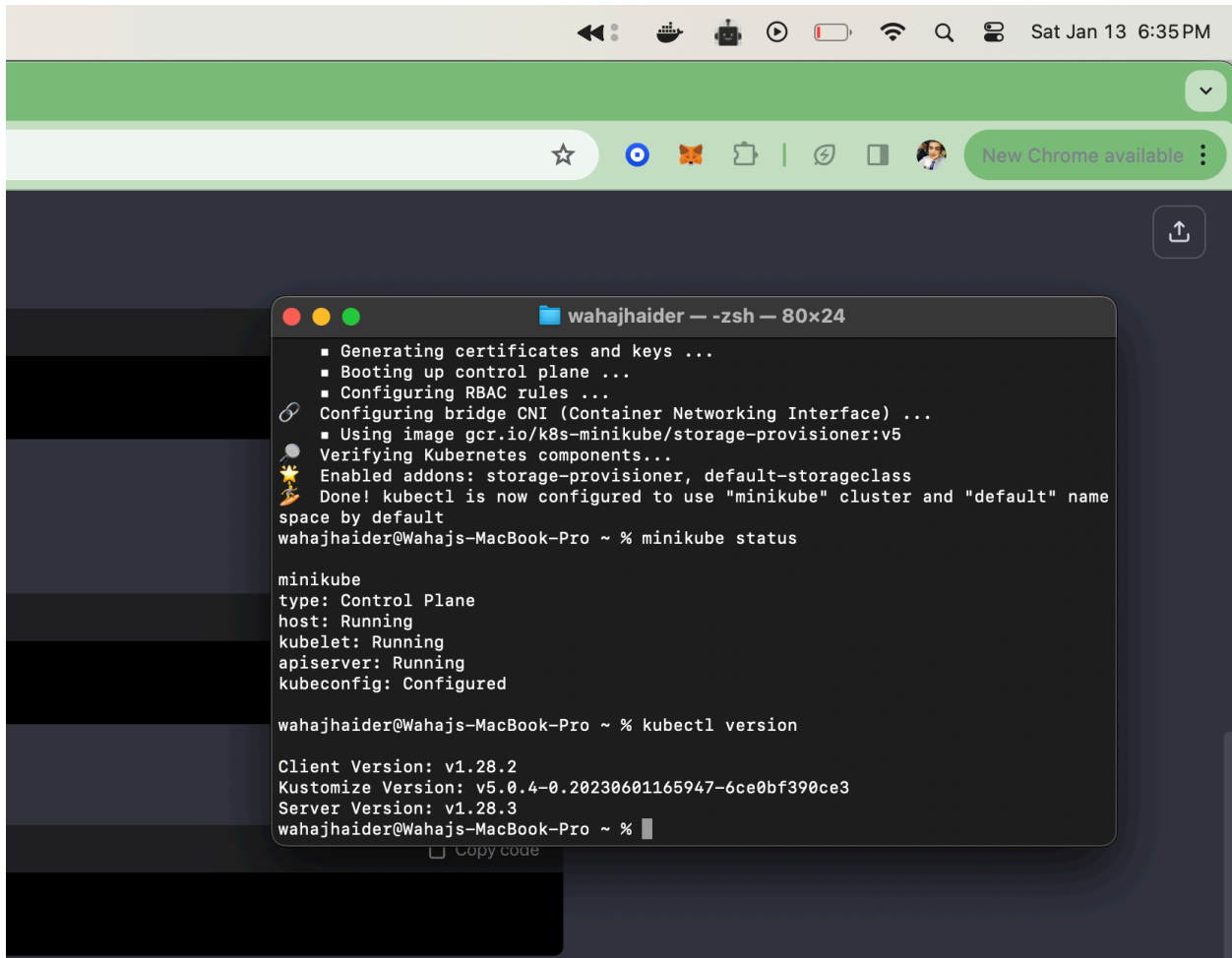


Kubect! Installed ->



The screenshot shows a Chrome browser window with a terminal window open. The terminal window title is "wahajhaider — zsh — 80x24". The terminal output shows the installation of minikube, including generating certificates, booting the control plane, and configuring RBAC rules. It also shows the status of minikube and the version of kubectl.

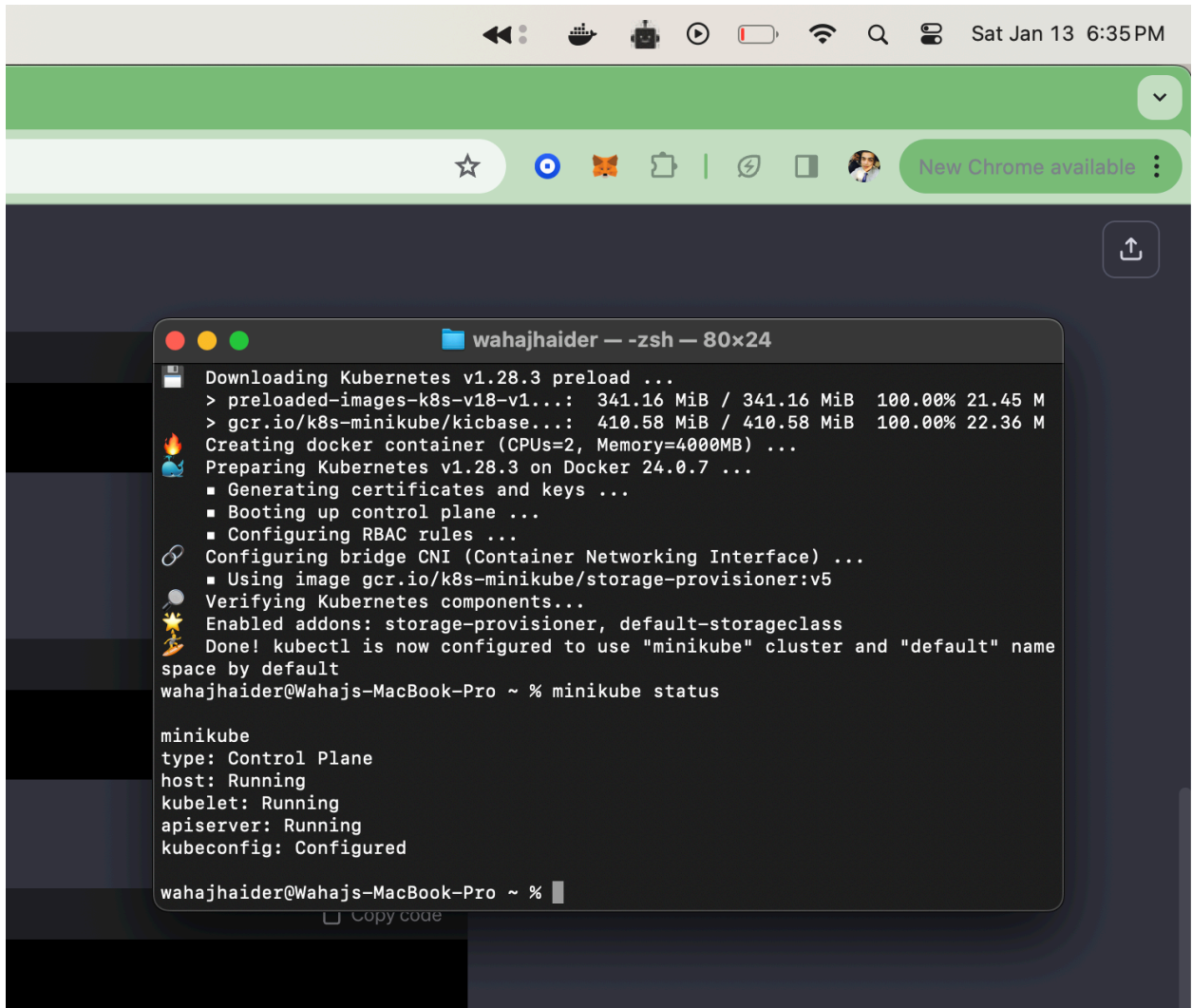
```
wahajhaider@Wahajs-MacBook-Pro ~ % minikube status

minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

wahajhaider@Wahajs-MacBook-Pro ~ % kubectl version

Client Version: v1.28.2
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
Server Version: v1.28.3
wahajhaider@Wahajs-MacBook-Pro ~ %
```

Minikube-→

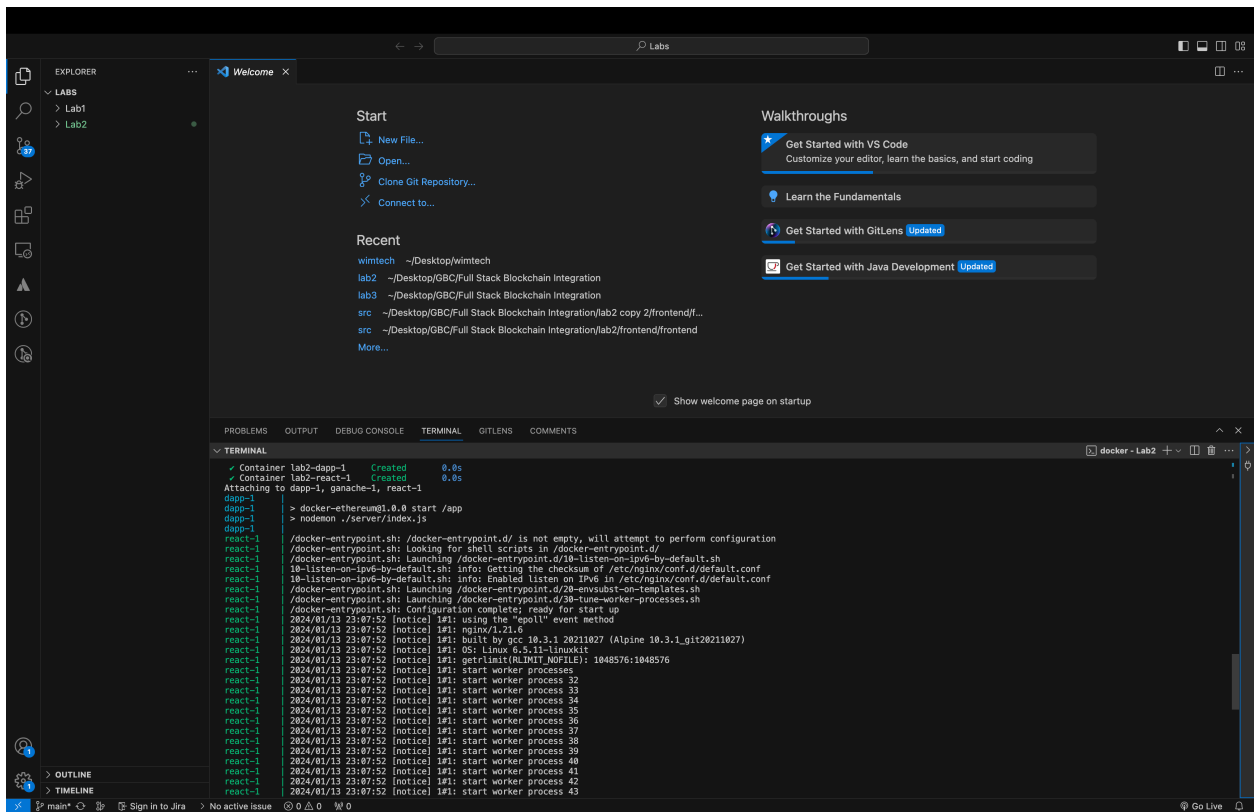


```
wahajhaider - zsh - 80x24
Downloading Kubernetes v1.28.3 preload ...
> preloaded-images-k8s-v18-v1...: 341.16 MiB / 341.16 MiB 100.00% 21.45 M
> gcr.io/k8s-minikube/kicbase...: 410.58 MiB / 410.58 MiB 100.00% 22.36 M
Creating docker container (CPUs=2, Memory=4000MB) ...
Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...
  ■ Generating certificates and keys ...
  ■ Booting up control plane ...
  ■ Configuring RBAC rules ...
Configuring bridge CNI (Container Networking Interface) ...
  ■ Using image gcr.io/k8s-minikube/storage-provisioner:v5
Verifying Kubernetes components...
Enabled addons: storage-provisioner, default-storageclass
Done! kubectl is now configured to use "minikube" cluster and "default" name
space by default
wahajhaider@Wahajs-MacBook-Pro ~ % minikube status

minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

wahajhaider@Wahajs-MacBook-Pro ~ %
```

Docker compose used to build and run the container for the dap



Ethereym dapp→

