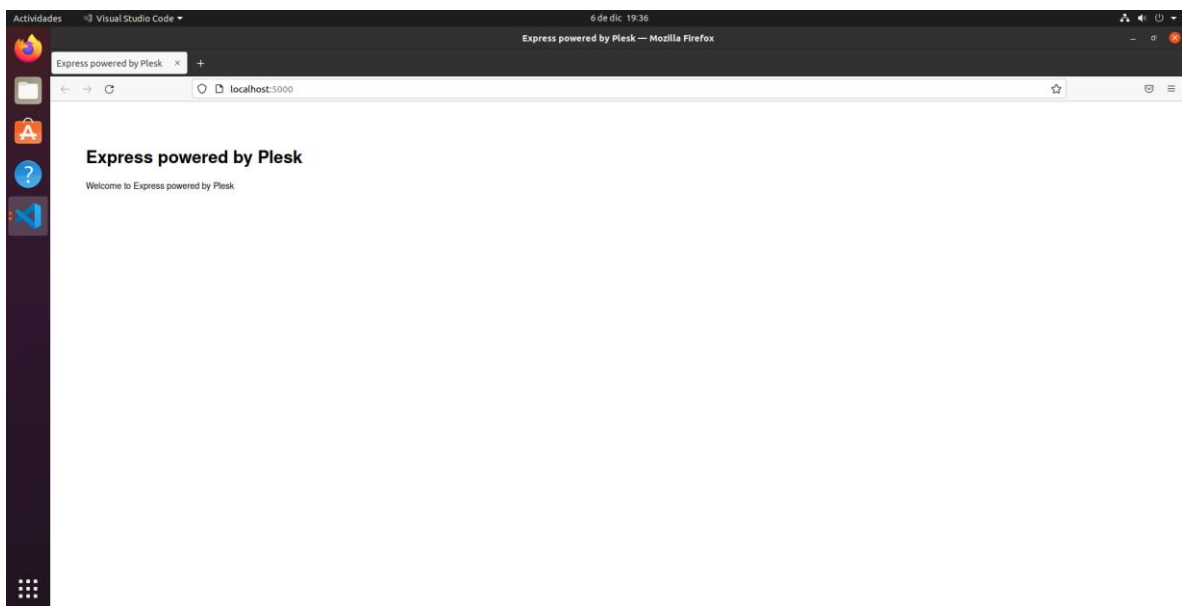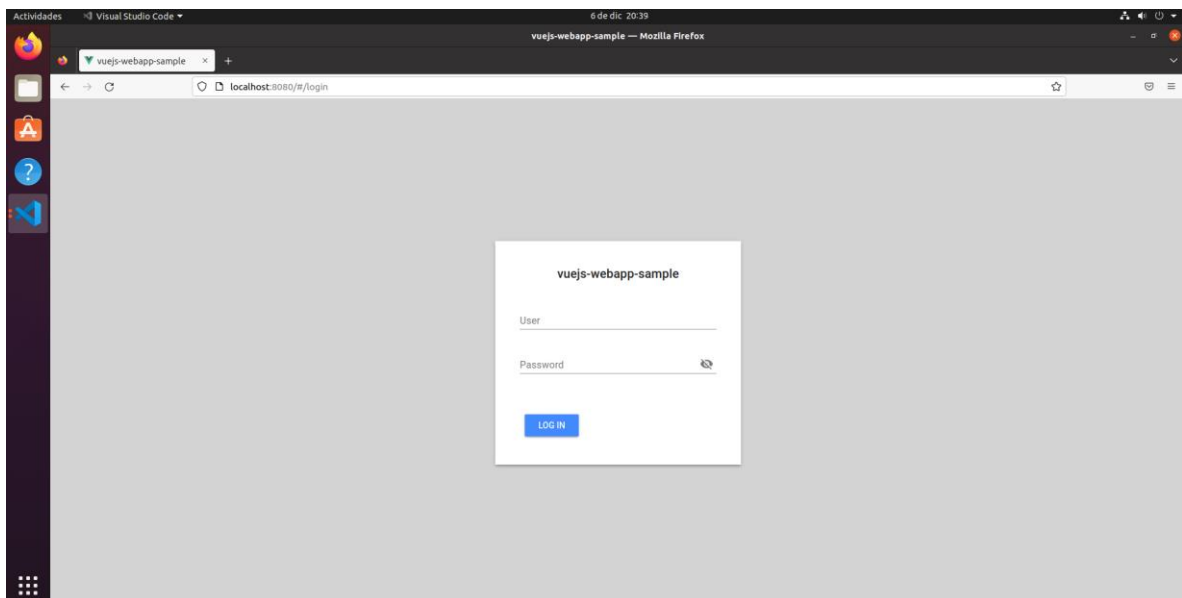# NodeJS

```dockerfile
FROM alpine
RUN apk add --update nodejs npm
WORKDIR /app
COPY package*.json ./
RUN npm ci
COPY . .
ENV PORT=5000
EXPOSE 5000
CMD ["npm", "start"]
```

Golang

```dockerfile
FROM golang:1.19 AS builder

WORKDIR /src
COPY . .
RUN go mod download
RUN CGO_ENABLED=1 GOOS=linux go build -o /app -a -ldflags '-linkmode external -extldflags "-static"' .

FROM scratch
COPY --from=builder /app /app
EXPOSE 8080

ENTRYPOINT ["/app"]
```
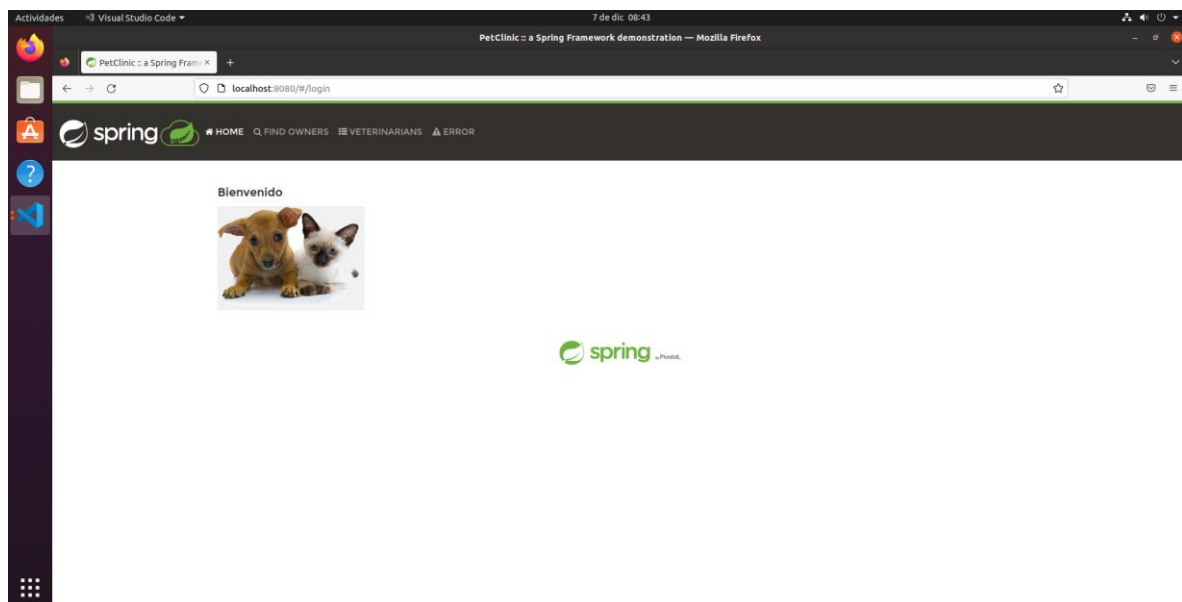
Java Spring


```
Dockerfile U ✕

Dockerfile > ...
  1    FROM adoptopenjdk/openjdk11:alpine-jre
  2
  3    ARG JAR_FILE=target/spring-petclinic-2.7.3.jar
  4
  5    WORKDIR /opt/app
  6
  7    COPY ${JAR_FILE} spring-petclinic-2.7.3.jar
  8
  9    EXPOSE 8080
 10
 11    ENTRYPOINT ["java","-jar","spring-petclinic-2.7.3.jar"]
```

Images Size

```
cristian@cristian-VirtualBox:~/Escritorio/spring-petclinic$ docker image ls
REPOSITORY              TAG          IMAGE ID        CREATED         SIZE
springpetclinic         latest       87763538d03a    4 minutes ago   202MB
gowebappsample          latest       58d0a2d17267    12 hours ago    36.1MB
nodeexpress             latest       67f6ca82c2ef    13 hours ago    91.2MB
```