##1 Automated Deployment

1) to run automated tests
dotnet build SuperService.UnitTests.csproj
dotnet test SuperService.UnitTests.csproj.dll --logger html

2) Docket file to create the image
FROM mcr.microsoft.com/dotnet/sdk:7.0 AS build
WORKDIR /app
COPY *.csproj ./
RUN dotnet restore
COPY . .
RUN dotnet publish -c Release -o out

FROM mcr.microsoft.com/dotnet/aspnet:7.0 AS runtime
WORKDIR /app
COPY --from=build /app/out ./
EXPOSE 80
ENTRYPOINT ["dotnet", "SuperService.dll"]

3) To create the image and container
docker build -t <image name> .
docker run <image name>

##2 Infrastructure

In AWS will create VPC, subnet (public and Private), IGW, NAT GWY, Route tables for private and public subnets, In public subnet route table will attach the route to IGW. Will launch instances in private and public subnets.

To provide the security will attach security group and NACL at instance level and Subnet level respectively. Will attach load balancer to distinguish the load evenly across multiple targets and autoscaler to adjust the load automatically based on traffic. For monitoring purpose will integrate cloudwatch to instances and will add the event and in target select SNS topic so if something happens then users will get automatically notified. For Orchestration purpose, will create the EKS cluster and will maintain this infra thought this cluster. and we can create Jenkins pipelines to automate the deployment process.