Background
A development team has created a Java web application that is ready for a limited release (with reduced availability and reliability requirements - till we get to production). If the limited release is successful, the app will be rolled out for worldwide use. Once fully public, the application needs to be available 24/7 and must provide sub-second response times and continuity through single server failures - targeting 4-9's availability.

Pre-requisites
As a devops engineer you need to devise a solution to cater below mentioned requirements.

- This assignment is not to test proficiency on any specific tool, but to understand the overall approach used to come up with a solution to this whole problem statement.
- Feel free to use any scripting/programming language to stitch the bits and pieces together.
- Use any Configuration management/Infra management/Virtualization tool of your choice to design the solution.

Problem Statement
The application will need to have HTTPS support through self signed certificates. You need to create one environment using the virtualization options provided below.

Assumptions
● You have free rein to incorporate any software tools and hardware you need to streamline the application deployment, infrastructure provisioning and configuration management as long as they are Free/Libre/Open Source software (FLOSS). We request that you use any flavor of Linux as the base OS.
● The development team has a continuous integration build that produces two artifacts:

  ○ a .zip file with static parts (images, stylesheets etc.) used for the application.
  ○ a .war file with the dynamic parts of the application.

● You should deploy the static assets to a web application and the war file to a separate application. Any compatible servers are acceptable.
● The app (Company News) uses Prevayler for persistence. Prevayler essentially persists data to a file. The dev team chose this to simplify the development effort, rather than having to deal with an RDBMS.

Expected output for this problem
Simply put, we want you to design and create the UAT/SIT environment, and provide a plan to scale out that deployment when the application goes public. You should use a virtualization solution consisting of any one of the following:

- Multi machine vagrant set up (static web and app in different VMs)
- Docker swarm (static web and app in different containers)
- Deploy in public cloud (AWS or AZURE), create static web and app in different instances and redirect network traffic between the two of them with an appropriate software of your choice.

You should provide executable scripts to enable us to build the environments ourselves.
In case you have another virtualization solution you would like to use please let us know in advance and we can discuss the same with you.

You are expected to deliver the following within 1 week:

- Scripts for creating the environment with the two applications as given in the problem statement.
- A plan for scaling the public release including hardware and software implications in form of a document. This can include diagrams and/or configuration scripts. If you see any issues scaling this application, please provide suggestions as a document that you might have for resolving them. This is a pretty open-ended problem.

- It will be great (but not mandatory!) if you can commit your scripts to a VCS (e.g. github/gitlab/bitbucket repository) and provide us read access to the same.
- Any tests written will definitely be considered a plus.

Out of scope

- Taking back ups
- Log aggregation
- Traffic flow management
- Application/Infra monitoring