- The steps involved in ~/alx-system_engineering-devops/0x09-web_infrastructure_deveps/0-simple_web_stack.jpg

- A user wants to access your website by typing www.foobar.com in their browser.
- The browser sends a DNS query to the domain name system, which resolves the domain name foobar.com to the server IP address 8.8.8.8
- The browser sends an HTTPS request to the server IP address, which is encrypted with the SSL certificate that you have installed on your server.
- The web server, which is Nginx, receives the HTTPS request and passes it to the application server, which handles the logic of your web application.
- The application server, which can be any software that runs your code, such as Python, Ruby, PHP, etc., executes the application files that contain your code base and queries the database if needed.
- The database, which is MySQL, stores and retrieves the data for your web application.
- The application server sends back the response to the web server, which formats it as HTML and sends it back to the browser.
- The browser displays the web page to the user.

- Some of the issues with this infrastructure.

  - SPOF stands for **single point of failure**, which means a part of a system that, if it fails, will stop the entire system from working. SPOFs are undesirable in any system with a goal of high availability or reliability[2]. For example, if you have only one web server and it crashes, your entire web application will be unavailable.
  - Downtime when maintenance needed means that your web application will be offline or inaccessible when you need to perform some tasks such as updating the code, installing patches, or restarting the web server. This can affect the user experience and satisfaction, as well as the security and performance of your web application.
  - Cannot scale if too much incoming traffic with one server infrastructure means that your web application will not be able to handle a large number of concurrent users or requests without compromising the response time or quality. This can lead to slow loading, errors, or crashes. Scaling is the ability of a system to accommodate larger loads by adding more resources, either by making the hardware stronger (scale up) or by adding more servers (scale out).