# Project Overview

The goal in this Project is to create Ansible playbooks for automation. This project will be the first step in creating what is known as a LAMP (Linux, Apache, MySQL, PHP) server. You’ll start by laying the framework down.

The goal of this project is to introduce you to roles, templates and handlers. Here’s a quick refresher on each of these.

**Roles**

Roles are a way to reuse functionality by putting all of the logic for a particular use case in the same place. As an example, if you have tasks that you might run on all servers regardless of what the server is used for, that logic might be put in a role called “common.”

And if you have logic to configure a web server, then you may have another role named “webserver.” And whenever you need to install a new web application, you can include these two roles in your playbook.

**Templates**

Templates at their core are a way to copy files to a remote server. However, the difference between templates and static files is that templates will be processed before they’re copied to the remote host. And this allows you to include variables, conditionals, loops, etc. Ansible uses Jinja2 as its template engine.

**Handlers**

Handlers are basically tasks that are run when some event happens. As an example, if you change a web server’s configuration file, you’ll need to restart the web server service. This is an ideal use case for handlers.

In this project, you will learn how to deploy a high-availability LAMP stack web app using AWS Elastic Beanstalk and Amazon Relational Database Service (RDS). The stack uses Linux, Apache, MySQL, and PHP. Using Elastic Beanstalk, you can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. Elastic Beanstalk automatically scales your application up and down based on your application's specific need using easily adjustable Auto Scaling settings. Amazon RDS makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while managing time-consuming database administration tasks.

# Project Architecture

